CODING OF IMAGE INTRA PREDICTION RESIDUALS USING SYMMETRIC GRAPHS

Alessandro Gnutti[⋆], *Fabrizio Guerrini*[⋆], *Riccardo Leonardi*[⋆] *and Antonio Ortega*[†]

[⋆] Department of Information Engineering, CNIT - University of Brescia, Italy
[†] Department of ECE, University of Southern California, Los Angeles, USA

ABSTRACT

The Discrete Cosine Transform (DCT) is widely deployed by modern image and video coding standards such as JPEG and H.26x. In most cases, the DCT is applied in a separable manner to rows and columns, which limits its ability to represent signals with diagonal orientation. As an alternative, non-separable transforms can represent signals with different orientations, but are significantly more computationally complex. To address this problem, in this paper we propose a set of non-separable Symmetry-Based Graph Fourier Transforms (SBGFTs), whose symmetric structures lead to a faster implementation. We study a practical image coding scenario that exploits the proposed SBGFTs, where for each intra predicted image residual block the optimal graph is chosen by solving a graph-based Rate-Distortion (*R-D*) problem. Experimental results indicate a coding efficiency higher than JPEG and JPEG2000.

Index Terms— Graph Fourier Transforms, symmetry, non-separable directional transforms, intra prediction image coding, fast implementation.

1. INTRODUCTION

Typically, the 2-D transform at the core of well-established image coding algorithms is separable. For example, both DCT in JPEG and wavelets in JPEG2000 are implemented in a separable manner along the vertical and horizontal directions. Separable transforms lead to lower computational complexity as compared to non-separable 2-D transforms, and provide good de-correlation for blocks with strong vertical or horizontal orientations. However, many image blocks may exhibit other types of directional correlation, e.g., due to the presence of oriented structures such as edges. Traditional separable 2-D transforms may not be able to efficiently represent these block types, since the corresponding eigenvectors do not provide orientations other than the vertical or horizontal one [1].

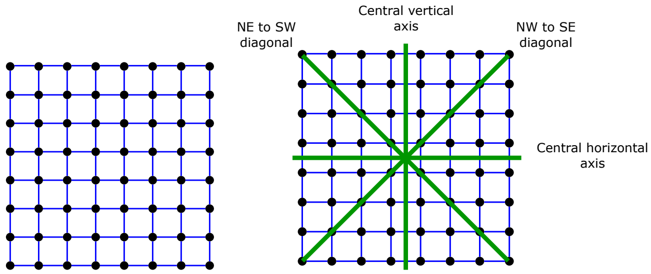
To overcome such a drawback, directional transforms have been proposed. Some approaches, such as Curvelets [2], Contourlets [3], Bandelets [4] or Directionlets [5], are more general, while others have focused specifically on directional transforms applied to image and video coding. For example, in [6] pixels in image blocks are re-ordered according to a given direction, and then classical transforms are applied. As an alternative, lifting-based methods are used to modify the conventional transforms to make them directional, as suggested in [7, 8] for image coding and in [9, 10] for video coding. In cases where directional prediction is used, this can be exploited so that a different directional transform is constructed based on the statistics of residual data within each prediction mode [11]. Some directional transforms are indeed separable, i.e., filtering is first performed along a dominant direction and then towards the orientation orthogonal to that. However, the separability constraint still necessarily reduces the directional selectivity.

Our proposed approach is based on Graph Signal Processing (GSP) [12, 13]. Graphs are models that represent complex interactions among data samples [14] by selecting edge weights between nodes. For a given graph, the Graph Fourier Transform (GFT) is often defined as the set of eigenvectors of the graph Laplacian matrix of that graph. If we consider an image processing application, an image block can be modeled as a graph with nodes as pixels and with different edge weights between nodes as a function of how correlated they are expected to be. Indeed, the DCT can be obtained as the GFT of a 2-D grid graph with equal unity weights (shown in Fig. 1a). In [15, 16, 17] different GFTs have been studied for the compression of inter and intra predicted video residuals and piecewise smooth images. Graph structure can also be learned from data. For example, in [18] graph learning problems are posed as the estimation of graph Laplacian matrices from some observed data.

In this paper, we propose a set of non-separable GFTs with directional bases that aim at achieving a sparse signal representation and thus higher image compression performance, while preserving low computational complexity. To that end symmetry properties in data, that are useful in a variety of signal processing tasks, e.g., [19, 20], can be exploited. In this work, symmetry is exploited to achieve highly efficient computation of the GFTs by restricting ourselves to pre-defined symmetric graphs. In particular, computation complexity lower than that typically obtained by non-separable transforms can be reached [21]. We will show that our proposed transforms can achieve a factor of 2 reduction in complexity with respect to non-separable approaches, and can achieve complexity comparable to that of separable transform when there exist more than one direction of symmetry in the graph (see results in Table 1).

In this paper, we extend our previous work [22], which only dealt with symmetric and near-symmetric graphs built on top of 2-D 4×4 grids, to the case of 8×8 image blocks. We focus exclusively on graphs having exact symmetry, and thus lower complexity implementation. We will show that our proposed Symmetry-Based GFTs (SBGFTs) have symmetry properties that lead to efficient representation of residual blocks with strong directional correlation. They are obtained by adding symmetric connections to the original DCT grid graph (see Fig. 1b). These symmetric connections in the graph will lead to transform basis functions having directional properties.

In what follows, we first show that allowing an encoder to select among multiple SBGFTs, each having different orientations, leads to better energy compaction than the DCT. Clearly, choosing among several possible transforms (rather than just using the DCT for all blocks) is key to the improved performance. Other approaches using multiple transforms, such as the mode dependent KLT [23], have been proposed in the past and have been shown to achieve better approximation as well. There are two significant differences between our approach and [23]: 1) our proposed transforms are not data-driven and are instead motivated by the existence of directional patterns on image blocks, and 2) their symmetry properties lead to faster



(a) 8×8 2-D grid graph with uniform weights.

(b) Different possible orientation of the symmetry axes. These structures also allow for GFT fast implementations.

Fig. 1: Preliminary concepts on symmetric graphs.

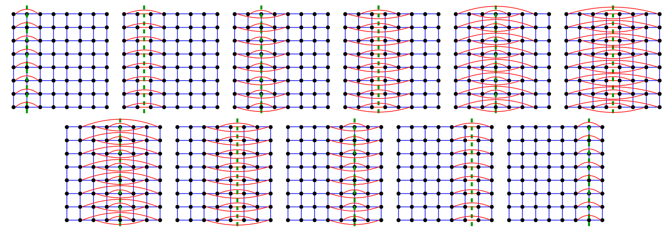
implementations, not available for KLTs. The preliminary study of the performance conducted in [22] was limited to the energy distribution in the transform domain. Instead, here we propose a practical image coding implementation using our SBGFTs. By comparing its rate-distortion (*R-D*) with respect to that of the most popular standard image coders (JPEG and JPEG2000), we prove that the proposed method achieves superior performance.

The rest of the paper is organized as follows. Section 2 defines the set of SBGFTs for the considered symmetric graphs. In Section 3 the approximation and compression performance that the SBGFTs lead to are described. Finally, conclusions are drawn in Section 4.

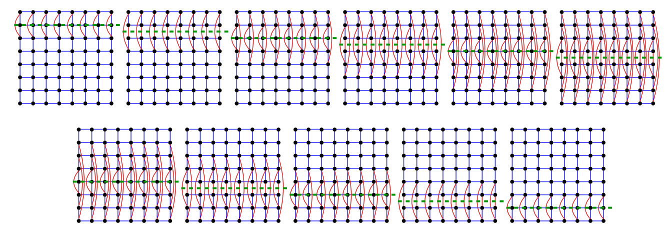
2. GFTS FOR SYMMETRIC GRAPHS

In graph signal processing, a GFT is a transform associated to a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathbf{W}\}$, where $\mathcal{V}$ and $\mathcal{E}$ represent the set of vertices and edges, respectively, and the matrix $\mathbf{W}$ is the adjacency matrix whose entry w_{ij} indicates the edge weight between the nodes i and j. Naming $\mathbf{D}$ the diagonal degree matrix of $\mathcal{G}$, the graph Laplacian matrix is defined as $\mathbf{L} = \mathbf{D} - \mathbf{W}$. Then, the GFT related to $\mathcal{G}$ is the matrix $\mathbf{U}$ of eigenvectors of $\mathbf{L}$. A graph signal $\mathbf{x}$ is a vector where each scalar entry is associated to one of the nodes of the graph. Then, the GFT of $\mathbf{x}$ is the vector of transform coefficients $\mathbf{y} = \mathbf{U}^\top \mathbf{x}$.

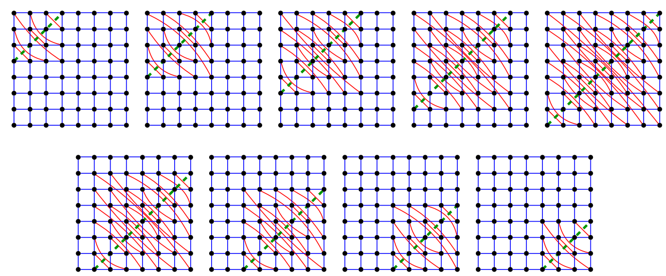
In this paper, a new set of symmetric graphs is constructed for 8×8 image blocks. The GFTs of graphs exhibiting symmetric structure will be called Symmetry-Based GFTs (SBGFTs). To design these graphs, we consider different reflection symmetries based on the angle and the position of the symmetry axis with respect to the data grid (Fig. 1). First, since we want to guarantee the preservation of the signal mean in the transform domain, we impose that each graph has to be fully connected. Indeed, if the graph is fully connected the eigenvalue $\lambda = 0$ of $\mathbf{L}$ will be simple, and the corresponding eigenvector will be the all 1 vector, which exactly matches the first basis function for the DCT. To guarantee a connected graph, our graph construction consists of a 2-D grid, with all the weights set to a constant value a, as shown in Fig. 1a, on top of which symmetric edges are added to incorporate symmetry into the graph (and thus orientation to the basis functions). This is done by selecting a symmetry axis and then adding edges perpendicular to it, so that nodes in specular position with respect to the symmetry axis are connected. Edge weights for these additional symmetric edges are set to a different constant value s. The graph parameters have been set to $a = 0.01$ and $s = 1$. Such ratio a/s has been experimentally found to be a reasonable choice, allowing to properly characterize the symmetry of graph signals.



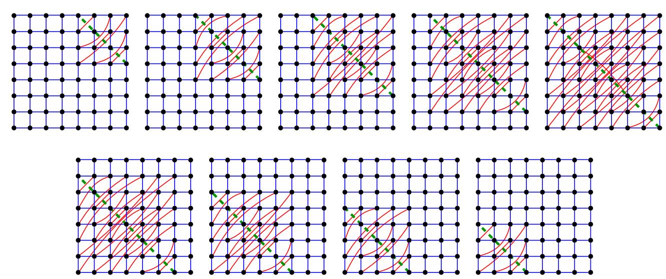
(a) Up-down symmetries.



(b) Left-right symmetries.



(c) Diagonal symmetries.



(d) Anti-diagonal symmetries.

Fig. 2: Proposed set of symmetric graphs. Blue indicates edges with weight set to $a = 0.01$, red denotes edges with weight set to $s = 1$. Green dashed segments mark the symmetry axes.

Due to the discrete nature of a graph, the slopes of the axes that better fit the grid are basically $0°$, $45°$, $90°$ and $135°$: those passing through the center of the grid are shown in Fig. 1b. For each axis orientation all the significant positions are evaluated. Specifically, the defined set contains graphs identified by the 40 following symmetries, depicted in Fig. 2:

(a) 11 up-down (UD) symmetries (Fig. 2a);

(b) 11 left-right (LR) symmetries (Fig. 2b);

(c) 9 diagonal (D) symmetries (Fig. 2c);

(d) 9 anti-diagonal (AD) symmetries (Fig. 2d).

For example, let us consider the second graph of Fig. 2b: the symmetry axis is marked as a green dashed line and it is placed between the

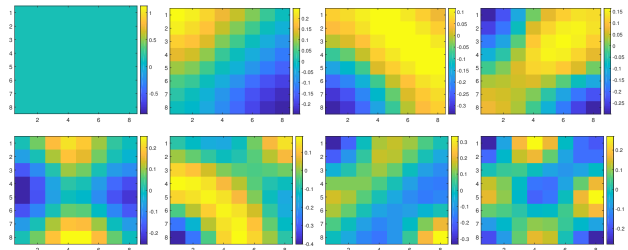


Fig. 3: 8 eigenvectors with smallest eigenvalues corresponding to the SBGFT of the 2nd graph in Fig. 2c.

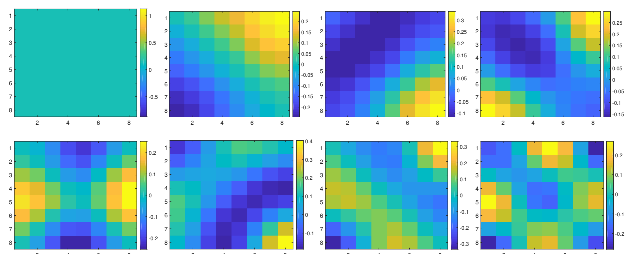


Fig. 4: 8 eigenvectors with smallest eigenvalues corresponding to the SBGFT of the 2nd graph in Fig. 2d.

second and the third column of the graph. Accordingly, the resulting edges connect the nodes in position $(2, i)$ with the ones in position $(3, i)$ and the nodes in position $(1, i)$ with the ones in position $(4, i)$, with $i = 1, \ldots, 8$.

To illustrate with an example, Figs. 3–4 show the first eight eigenvectors corresponding to the SBGFTs of the second graph in Fig. 2c and the second one in Fig. 2d, respectively. These figures are particularly interesting because they indicate how the bases can achieve orientation also in D- and AD-symmetries where such symmetries cannot be obtained with separable transforms.

A property of all our proposed graphs is that their respective GFTs admit computational speedup techniques. In [21], the authors have shown that butterfly-based speedup methods can be exploited when all the edges and self loops of $\mathcal{G}$ are symmetric about the structures shown in Fig. 1b, that is:

1. central horizontal axis;
2. central vertical axis;
3. the northwest-to-southeast diagonal;
4. the northeast-to-southwest diagonal.

We can prove that the graphs in (a), (b), (c) and (d) are symmetric with respect to the axis indicated in 1), 2), 3) and 4), respectively. Indeed, let $(e_{i,j})_{k,l}$ be the edge associated to the k-th node of column i and the l-th node of column j. Then, it is straightforward to observe that:

- $(e_{i,j})_{k,l} = (e_{i,j})_{N+1-k,N+1-l}$ for graphs in (a);
- $(e_{i,j})_{k,l} = (e_{N+1-i,N+1-j})_{k,l}$ for graphs in (b);
- $(e_{i,j})_{k,l} = (e_{k,l})_{i,j}$ for graphs in (c);
- $(e_{i,j})_{k,l} = (e_{N+1-k,N+1-l})_{N+1-i,N+1-j}$ for graphs in (d).

Since these conditions satisfy the symmetry requirements with respect to the symmetry axis as described by 1), 2), 3) and 4), respectively, then all the graphs in Fig. 2 admit a fast implementation. Note that the sixth graphs in Figs. 2a and 2b are simultaneously UD- and LR-symmetric, whereas the fifth graph in both Fig. 2c and 2d are simultaneously D- and AD-symmetric. The number of multiplications for the symmetric graph types described above are summarized in Table 1. The reader is referred to [21] for more details on how lower complexity can be achieved for the various types of symmetries considered here.

Grid type	Number of multiplications
UD-symmetry	$N^4/2$
LR-symmetry	$N^4/2$
D-symmetry	$N^2(N^2+1)/2$
AD-symmetry	$N^2(N^2+1)/2$
Both UD- and LR-symmetry	$N^4/4 \stackrel{N\lesssim 9}{\sim} 2N^3$
Both D- and AD-symmetry	$N^2(N^2+1)/4 \stackrel{N\lesssim 9}{\sim} 2N^3$
Non-separable, no symmetry	N^4
Separable, no symmetry	$2N^3$

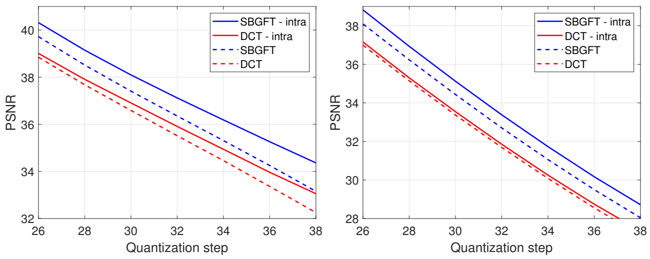
Table 1: Types of grid symmetries and the number of multiplications to compute the corresponding speeded up GFTs, compared with the cost of separable and non-separable transforms (last two rows).

3. EXPERIMENTAL RESULTS

First, in order to evaluate the approximation ability of the SBGFTs associated to the graphs in Fig. 2 with respect to the DCT, a PSNR-quantization analysis is presented. The experiments have been carried out on a variety of standard images obtained from the USC-SIPI Image Database [24]. Two types of procedures have been followed. In the first case, the transforms have been applied directly on the original image. In the second scenario, the DCT and the SBGFTs have been applied on the image residual blocks obtained after intra prediction. The residuals are calculated by considering the 35 prediction modes adopted for intra-frame prediction in H.265 [25, 26].

Basically, a "brute-force" strategy is adopted in which all the 40 SBGFTs are tested for each 8×8 block. Note that an exhaustive search is computationally inefficient, but this does not affect the decoding stage where the complexity still remains low. Furthermore, recently proposed methods can reduce transform selection by exploiting the graph representation [27]. We will include these speed up techniques in future work.

Transform coefficients are quantized with different quantization steps, and the corresponding de-quantized blocks are inverse-transformed by using the inverse-SBGFTs. The graph leading to the smallest mean squared error (MSE) is selected as the optimal graph used for that block. The same method is used for both of the two aforementioned procedures. The only difference is that when intra prediction is considered, the search for the optimal graph is combined with the investigation of the best prediction mode. Finally, the peak signal-to-noise ratio (PSNR) corresponding to the error between the original and the reconstructed images is used to



(a) *Lena*. (b) *Baboon*.

Fig. 5: PSNR-quantization curves for selected images. Dashed lines indicate that the transforms are applied on the original image blocks, whereas solid lines signify that the transforms are applied on the intra predicted residual blocks.

quantify quality.

In Fig. 5 performance curves are shown for the *Lena* and *Baboon* images, both of size 512×512[1]. Results show our proposed symmetric graphs can outperform the DCT in terms of approximation ability. Interestingly, even when the SBGFTs are applied directly on the original image blocks, the achieved PSNR values are still higher than the ones obtained by using the DCT after intra prediction. Note that no bits overhead has been taken into account to choose the transform. Indeed, these results are intended to remark the quality of the graph transforms in terms of energy compaction, without considering a realistic image coding scenario.

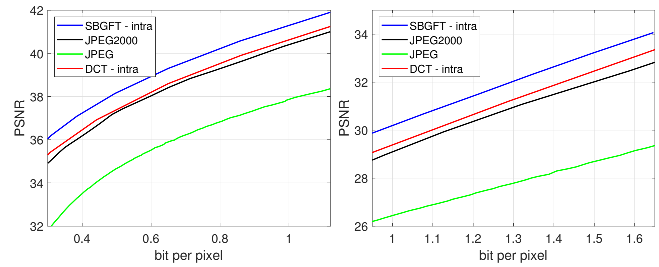
The second set of experiments instead compares the real coding performance of the proposed SBGFTs and two popular standard image coders: JPEG and JPEG2000. Since Fig. 5 has shown a higher performance when the SBGFTs are applied following intra prediction, then in these next experiments the graph transforms will be applied on intra prediction residuals. Since JPEG and JPEG2000 do not use intra prediction, the curve obtained using intra predicted DCT is reported as well to prove that the performance gain of the proposed transforms is not just due to the employment of intra prediction. The prediction mode and the prediction error signal (i.e., the quantized coefficients) are transmitted to the decoder, as for intra frame prediction in H.265. This information is binarized and entropy coded using Context-Based Adaptive Binary Coding (CABAC) [28], introduced in H.264/AVC and now used in the latest HEVC standard as well. In order to signal the index of the graph used for each block, a fixed-length bit sequence is sent. Note that this actually represents the worst-case graph index coding scenario, i.e., we do not consider how the selection of the optimal graph in a given block is potentially correlated to the graphs and prediction modes selected in neighboring blocks, and in addition there is no entropy coding exploiting the probability distribution of SBGFTs for a given mode. Thus, each graph is represented by a sequence of $\lceil \log_2 40 \rceil = 6$ bits. The analysis for a better graph index representation is subject of our current studies.

The choice of coding parameters is based on *R-D* optimization. For each block an unconstrained optimization problem is solved, namely:

$$\min_{\text{(coding parameters and graphs)}} J_{\mathcal{G}} = D_{\mathcal{G}} + \lambda \cdot R_{\mathcal{G}} \qquad (1)$$

where $J_{\mathcal{G}}$ is the Lagrangian cost function, $D_{\mathcal{G}}$ is the distortion, λ is a quantization-based non-negative Lagrangian multiplier and $R_{\mathcal{G}}$ is the rate, i.e., the number of bits required to signal the prediction

[1]The results shown in this section are consistent with those that have been obtained on other images in the considered dataset.



(a) *Lena*. (b) *Baboon*.

Fig. 6: *R-D* curves for selected images.

	Lena			*Baboon*		
	J2K	JPG	DCT intra	J2K	JPG	DCT intra
Δ PSNR	1	3.4	0.76	1.2	3.92	0.8
Δ rate	-18.2%	-53.5%	-15.5%	-15.3%	-47%	-10.3%

Table 2: *BD* rate associated to Fig. 6.

mode and the quantized coefficients. Clearly, the selection of the optimal SBGFT has a strong effect on both the distortion and rate.

In Fig. 6 the performance of our set of transforms is illustrated. Each *R-D* point is computed by calculating the PSNR between the original and the reconstructed images for a fixed quantization step. The curves show that the SBGFTs outperform both JPEG and JPEG2000, as well as DCT with intra prediction, even considering 6 overhead bits for each block to signal the optimal graph index. In Table 2 the corresponding Bjøntegaard's metric (*BD* rate) is also reported in order to show the average gain in terms of PSNR and the average bit rate saving percentage between the compared rate-distortion curves. In the first row a positive value indicates a PSNR increase for the same bit rate, whereas in the second row a negative value identifies a decrease of bit rate for the same PSNR. For example, in *Lena* the set of SBGFTs has an average gain of 1 dB and it requires -18.2% of bit rate on average with respect to JPEG2000.

4. CONCLUSIONS

In this paper, we have proposed a new set of transforms based on graphs for image coding. These graphs are built on 8×8 grids and have some specific symmetric configurations. We have shown that the corresponding Symmetry-Based Graph Fourier Transforms (SBGFTs) have a fast implementation, making the proposed transforms useful in a practical image coding scenario. We have compared this set of transforms with the DCT both on natural images and intra predicted image residuals. The experimental results indicate that the SBGFTs outperform the DCT in terms of energy compaction, thanks to its ability to represent signals exhibiting oriented structures. Furthermore, a graph-based image coder has been implemented and it shows performance higher than JPEG and JPEG2000.

To signal the graph index used for each block, a fixed-length sequence bit is sent. This is a worst-case scenario in which no correlation is supposed to exist between the optimal graph chosen for a given block and the graphs and prediction modes selected in neighboring blocks. Future work will be focused on studying potential dependencies between selected graphs in order to obtain a more efficient graph signaling coding, and introducing some additional speed up techniques.

5. REFERENCES

[1] J. Xu, B. Zeng, and F. Wu, "An overview of directional transforms in image coding," in *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*, May 2010, pp. 3036–3039.

[2] Emmanuel Candès and David Donoho, "Curvelets - a surprisingly effective nonadaptive representation for objects with edges," *Curves and Surfaces*, April 2000.

[3] M. N. Do and M. Vetterli, "The contourlet transform: an efficient directional multiresolution image representation," *IEEE Transactions on Image Processing*, vol. 14, no. 12, pp. 2091–2106, Dec 2005.

[4] E. Le Pennec and S. Mallat, "Sparse geometric image representations with bandelets," *IEEE Transactions on Image Processing*, vol. 14, no. 4, pp. 423–438, April 2005.

[5] V. Velisavljevic, B. Beferull-Lozano, M. Vetterli, and P. L. Dragotti, "Directionlets: anisotropic multidirectional representation with separable filtering," *IEEE Transactions on Image Processing*, vol. 15, no. 7, pp. 1916–1933, July 2006.

[6] B. Zeng and J. Fu, "Directional discrete cosine transforms: A new framework for image coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 3, pp. 305–313, March 2008.

[7] G. Shen and A. Ortega, "Compact image representation using wavelet lifting along arbitrary trees," in *2008 15th IEEE International Conference on Image Processing*, Oct 2008, pp. 2808–2811.

[8] W. Ding, F. Wu, X. Wu, S. Li, and H. Li, "Adaptive directional lifting-based wavelet transform for image coding," *IEEE Transactions on Image Processing*, vol. 16, no. 2, pp. 416–427, Feb 2007.

[9] E. Martinez-Enriquez, J. Cid-Sueiro, F. Daz de Mara, and A. Ortega, "Directional transforms for video coding based on lifting on graphs," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 4, pp. 933–946, April 2018.

[10] Y. Chao, A. Ortega, and S. Yea, "Graph-based lifting transform for intra-predicted video coding," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2016, pp. 1140–1144.

[11] Y. Ye and M. Karczewicz, "Improved h.264 intra coding based on bi-directional intra prediction, directional transform, and adaptive coefficient scanning," in *2008 15th IEEE International Conference on Image Processing*, Oct 2008, pp. 2116–2119.

[12] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 83–98, May 2013.

[13] A. Ortega, P. Frossard, J. Kovaevi, J. M. F. Moura, and P. Vandergheynst, "Graph signal processing: Overview, challenges, and applications," *Proceedings of the IEEE*, vol. 106, no. 5, pp. 808–828, May 2018.

[14] Leo J. Grady and Jonathan R. Polimeni, *Discrete Calculus: Applied Analysis on Graphs for Computational Science*, Springer Publishing Company, Incorporated, 1st edition, 2010.

[15] H. E. Egilmez, A. Said, Y. H. Chao, and A. Ortega, "Graph-based transforms for inter predicted video coding," in *2015 IEEE International Conference on Image Processing (ICIP)*, Sept 2015, pp. 3992–3996.

[16] W. Hu, G. Cheung, and A. Ortega, "Intra-prediction and generalized graph fourier transform for image coding," *IEEE Signal Processing Letters*, vol. 22, no. 11, pp. 1913–1917, Nov 2015.

[17] Wei Hu, Gene Cheung, Antonio Ortega, and Oscar C. Au, "Multiresolution graph fourier transform for compression of piecewise smooth images," *IEEE Trans. Image Processing*, vol. 24, no. 1, pp. 419–433, 2015.

[18] H. E. Egilmez, E. Pavez, and A. Ortega, "Graph learning from data under laplacian and structural constraints," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 6, pp. 825–841, Sept 2017.

[19] A. Gnutti, F. Guerrini, and R. Leonardi, "Representation of signals by local symmetry decomposition," in *Proc. of the 23rd European Signal Processing Conference (EUSIPCO '15)*, 2015, pp. 983–987.

[20] Fabrizio Guerrini, Alessandro Gnutti, and Riccardo Leonardi, "Even/odd decomposition made sparse: A fingerprint to hidden patterns," *Signal Processing*, vol. 141, pp. 273 – 286, 2017.

[21] Keng-Shih Lu and Antonio Ortega, "Fast implementation for symmetric non-separable transforms based on grids," *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4109–4113, 2017.

[22] A. Gnutti, F. Guerrini, R. Leonardi, and A. Ortega, "Symmetry-based graph fourier transforms for image representation," in *2018 25th IEEE International Conference on Image Processing (ICIP)*, Oct 2018, pp. 2575–2579.

[23] C. Yeo, Y. H. Tan, and Z. Li, "Low-complexity mode-dependent klt for block-based intra coding," in *2011 18th IEEE International Conference on Image Processing*, Sep. 2011, pp. 3685–3688.

[24] "The USC-SIPI image database," http://sipi.usc.edu/database/.

[25] G. J. Sullivan, J. Ohm, W. Han, and T. Wiegand, "Overview of the high efficiency video coding (hevc) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, Dec 2012.

[26] J. Lainema, F. Bossen, W. Han, J. Min, and K. Ugur, "Intra coding of the hevc standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1792–1801, Dec 2012.

[27] K. Lu, A. Ortega, D. Mukherjee, and Y. Chen, "Efficient rate-distortion approximation and transform type selection using laplacian operators," in *2018 Picture Coding Symposium (PCS)*, June 2018, pp. 76–80.

[28] D. Marpe, H. Schwarz, and T. Wiegand, "Context-based adaptive binary arithmetic coding in the h.264/avc video compression standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 620–636, July 2003.