DIRECTION-ADAPTIVE TRANSFORMS FOR IMAGE COMMUNICATION

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF ELECTRICAL ENGINEERING
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Chuo-Ling Chang
June 2009

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Prof. Bernd Girod) Principal Adviser

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Prof. Robert Gray)

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Dr.-Ing. Markus Flierl)

Approved for the University Committee on Graduate Studies.

Abstract

This dissertation investigates direction-adaptive transforms that adapt transform directions to local directionality in images in order to improve the performance of image coding. Two new direction-adaptive transforms are proposed: the direction-adaptive partitioned block transform (DA-PBT) and the direction-adaptive discrete wavelet transform (DA-DWT), together with a novel image model that facilitates theoretical analysis of the coding performance.

The image model is able to represent locally varying textures in the image, hence suitable for the analysis of adaptive transforms. Using the model, theoretical analysis shows that significant improvements can be expected around sharp directional features such as edges and lines, rendered by directional bases of the adaptive transform.

To provide these directional bases in practice, the DA-PBT is proposed based on the DCT for coding of image and video sequences. It outperforms the conventional 2-D DCT in rate-distortion performance as well as visual quality. It can also be combined with block-based predictive coding, both directional prediction within the image or interframe prediction for video to further improve the compression efficiency.

For coding of still images, the DA-DWT is proposed based on the DWT. Although the directions are selected block-wise, using the lifting structure, filtering extends across block boundaries such that, unlike the DA-PBT, inter-block correlation can be exploited and blocking artifacts are absent in the reconstruction. At low rates, different from the ringing and checkerboard artifacts of the conventional transforms, the DA-DWT results in brushstroke-like artifacts that better preserve the geometric structure in the image, providing a visually more pleasing image representation.

Acknowledgment

I would like to thank my advisor, Professor Bernd Girod, for his valuable guidance during my doctoral program. Other members of my reading committee, Professor Robert Gray and Dr. Markus Flierl, have also made numerous suggestions that improved my doctoral research and this dissertation. I would also like to extend my sincere gratitude to the members, alumni, visitors, friends, and assistants of the Image, Video and Multimedia Systems group, and the colleagues in the Information Systems Laboratory at Stanford University for their stimulating discussions and their friendship throughout the years. Most importantly, I would like to thank my family members, especially my parents, for their constant love and support. This dissertation is dedicated to them.

Contents

List of Figures

Chapter 1

Introduction

The quest for efficient image coding is an on-going research effort of both theoretical and practical interest. Among image coding techniques, transform coding is recognized as one of the most successful methods, playing a crucial role in almost all of the state-of-the-art image and video coding standards. This success in modern applications is in fact built upon decades of studies. In this chapter, we first review early developments of transform coding of images in Sec. 1.1, and in particular examine two transforms widely adopted in practice: the discrete cosine transform (DCT) and the discrete wavelet transform (DWT). In two dimensions, the DCT and the DWT are not able to efficiently represent directional image features not aligned vertically or horizontally, due to the lack of directional bases in such orientations. In Sec. 1.2, we review works on directional transforms that provide these directional bases. The directional transforms are very effective in image processing tasks, however, their performance for image coding is rather unsatisfactory. Another group of transforms that adapt the choice of directional bases according to local directionality in images is presented in Sec. 1.3. These direction-adaptive transforms can achieve substantial improvement in coding performance, both objectively and subjectively, over the conventional non-directional transforms, and they constitute the main theme of this dissertation. A summary of the research contributions in the dissertation is provided in Sec. 1.4, and the organization of the remaining chapters is described in Sec. 1.5.

1.1 Transform Coding of Images

A general framework of transform coding consists of three steps: an invertible linear transformation that converts a vector of source samples into a vector of coefficients, independent quantization optimized for each scalar coefficient, and entropy coding that maps the quantization indices into a bitstream. To reconstruct the source samples, the bitstream is first decoded losslessly to obtain the quantization indices, followed by inverse quantization that recovers the coefficients, possibly with loss in fidelity, and finally the inverse transform. This framework divides the challenging task of designing optimal codes for the source into a sequence of three simpler steps. By first exploiting the correlation among the source samples with the linear transform, scalar quantization and simple entropy coding may be applied to the decorrelated coefficients without significant loss in coding performance [88].

Among all linear transforms with the same dimension, the Karhunen Loève Transform (KLT) generates uncorrelated coefficients for a given source and achieves maximum energy compaction by packing the most energy in any given number of coefficients. To be specific, if in the inverse transform only a given number of coefficients with the largest magnitude are kept and the remaining are set to zero, the KLT is the transform with the minimum mean-squared-error (MSE) distortion in the reconstruction [14, 106]. Furthermore, for source coding, assuming jointly Gaussian source samples and independent quantization and entropy coding of each scalar coefficient at high rates, the KLT achieves the minimum MSE distortion in the reconstruction among all orthogonal transforms at the same rate [88, 106].

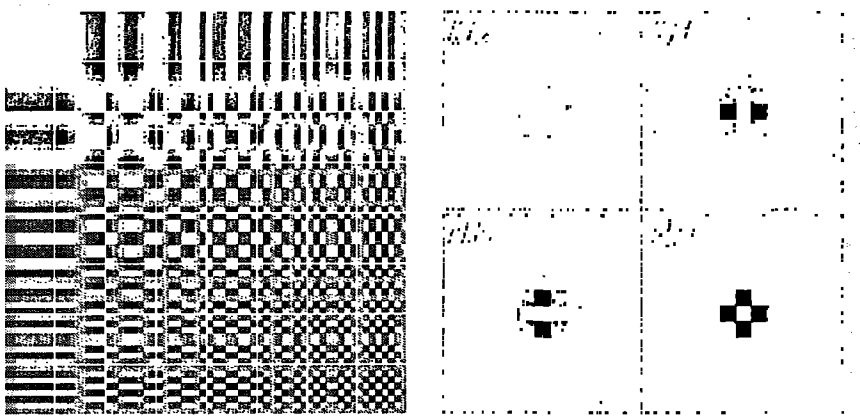
Exploiting the energy compaction property of the KLT, Kramer and Mathews [117] first developed a system for the transmission of highly-correlated continuous-time speech signals. Instead of sending the whole set of signals, only a few linear combinations of the signals computed based on the KLT are transmitted, achieving bandwidth reduction with a prescribed fidelity. For discrete-time signals, Huang and Schultheiss [100, 101] first proposed the aforementioned transform coding framework, analyzed the performance assuming jointly Gaussian source samples and fixed-rate scalar quantization, and concluded that the KLT is optimal for the task. Despite its

theoretical optimality, in practice the KLT of a given source is generally difficult to compute and implement.

For practical coding systems, several transforms designed to approximate the performance of the KLT with efficient implementation have been proposed. For instance, the Hadamard transform was proposed for coding of speech signals [45,86]. Transform coding of images was first proposed by Andrews and Pratt [6] using fast implementation of the discrete Fourier transform (DFT) [41,43], applied to the entire image in a separable fashion, i.e., applying 1-D transforms vertically along columns of image samples followed by further processing of the resulting coefficients with horizontal 1-D transforms along rows of the coefficients. The authors later adopted the Hadamard transform [7, 156], again applied as a separable transform to 2-D images. Instead of transforming the entire image, transform coding can be applied to image blocks, lowering the complexity of the transform and allowing more flexibility in quantization and entropy coding. For example, the method presented in [5] applies the DFT to image blocks composed of 16×16 samples and adaptively selects quantization accuracies for different types of blocks. The performance of three block transforms, the KLT, the DFT and the Hadamard transform, is analyzed and compared in [98]. The Slant transform was introduced in [69] and was later extended to use a fast construction [155], demonstrating improved performance over the DFT and the Hadamard transform. Detailed reviews of early developments of transform image coding can be found in [37, 105, 143, 208].

1.1.1 Discrete Cosine Transform

Since its introduction in [2], the DCT has become the main focus of transform coding of image blocks for its ability to better approximate the KLT for smooth regions in images than the previously proposed block transforms, observed both from experiments [160] and via theoretical justification [38, 99]. For image blocks, the DCT is applied as a separable 2-D transform, possibly with fast implementation [34, 120]. The 64 basis functions of the 8×8 DCT are shown in Fig. 1.1-(a). Each basis function is associated with a DCT coefficient, representing an image block as a linear



(a) (b)

Figure 1.1: (a) Basis functions of the 8×8 DCT, (b) basis functions of one level of the separable 2-D DWT (forward transform) using the 9/7 filter pair.

combination of the bases. The DCT is adopted as the core of the JPEG image coding standard, which was initiated in 1986 and issued in 1992 [109, 150, 198, 199].

The DCT is also widely adopted in the hybrid coding framework for images and image sequences. In hybrid coding of images [97], block transforms are first applied to the image and the transform coefficients are encoded with DPCM, i.e., using prediction generated from the reconstruction of the corresponding coefficients in previously encoded blocks. For image sequences, the prediction signal can be derived from perviously encoded images in the sequence to further exploit the temporal correlation [165]. In [107], motion-compensated interframe prediction is incorporated into the DPCM step of hybrid coding to better explore the temporal redundancy. The authors have also suggested an alternative structure of hybrid coding that first performs the DPCM step, i.e., motion-compensated prediction, followed by transform coding of the prediction residual. This structure has become the common framework for most video coding standards, for instance, H.263 [162], H.263+ [78], and most recently H.264 [206], all using the DCT or its simplified variant.

Despite its ability to decorrelate samples in image blocks, as a block transform, the DCT cannot exploit the correlation across block boundaries. Additionally, at lower rates, independent processing of image blocks creates blocking artifacts that can only be partially mitigated with post-processing algorithms [172], substantially

deteriorating visual quality. To overcome these limitations, based on the DCT, Malvar and Staelin proposed the lapped transform [136–139] that operates on overlapped blocks while overall remaining a critically sampled transform, i.e., generating the same number of coefficients as the number of samples in the image. The work in [187] further shows that a large family of the lapped transforms can be constructed by adding simple pre-processing and post-processing components to the DCT-based framework with disjoint blocks. This approach is adopted in HD Photo [176], a recently developed image coding system for applications in digital photography.

1.1.2 Discrete Wavelet Transform

Another image coding technique free from blocking artifacts is subband coding [195, 197, 203, 209], also conforming with the transform coding framework. In subband coding, the transform is implemented through a set of filtering and subsampling procedures applied to the entire image. The set of filters is designed to have approximately non-overlapping frequency responses, hence decomposing the source samples into uncorrelated frequency components, and subsampling enables a critically sampled transform. In this regard, block transforms such as the DCT can be considered as a special class of subband transforms, with the support of the filters, i.e., the bases, limited within a block. For general subband transforms, the filters are constructed with certain constraints such that the source samples can be perfectly reconstructed from the coefficients by another set of upsampling and filtering operations, where the set of filters used in the forward and the inverse transform may be different, leading to a biorthogonal transform [188].

The discrete wavelet transform (DWT) [8, 9, 124, 135] is also a special case of subband transforms. The filters used in the DWT are typically designed based on certain smoothness constraints [42, 46, 134]. The DWT decomposes the source into a low-pass (L) and a high-pass (H) subband. In two dimensions, the separable 2-D DWT leads to four subbands, LL, LH, HL and HH, in one level of the 2-D transform. The basis functions of the separable 2-D DWT using the popular 9/7 filter pair [42,47] are shown in Fig. 1.1-(b). They correspond to the subband filters resulting in the

four subbands in the forward transform. The decomposition can be continued by recursively applying the process to the subsampled low-resolution image, i.e., the LL subband, providing a critically sampled yet multi-resolution representation.

To encode wavelet coefficients, Shapiro proposed the EZW algorithm [171] that exploits the remaining correlation across resolutions, and reported rate-distortion results outperforming than the JPEG standard. More importantly, EZW generates an embedded bitstream, i.e., different qualities of the reconstruction can be obtained by truncating the bitstream at different lengths. The performance of embedded entropy coding have been further improved in, for instance, [166] and [183]. In [179], Sweldens showed that the DWT can be implemented using the lifting structure. By factoring the DWT into pairs of lifting steps, the lifting structure enables an efficient construction of the DWT and facilitates the design of a new variety of wavelet filters [16, 47].

Because of the absence of blocking artifacts, multi-resolution representation and superior compression performance, the separable 2-D DWT has been adopted in the JPEG2000 image coding standard defined in 2000 [110], which not only outperforms JPEG in compression performance but also provides more functionalities [36,175,185]. As a side remark, the performance of DCT-based image coding has been greatly improved after the introduction of the JPEG standard. For example, an extension of H.264 using the hybrid coding principle for still images [141] and other schemes using the lapped transform with more efficient embedded entropy coding [170, 176] have shown performance comparable to JPEG2000 [48].

1.2 Directional Transforms

Due to the separable structure that constructs a 2-D transform by cascading a 1-D vertical transform with a 1-D horizontal transform, the basis functions of the 2-D DCT and the separable 2-D DWT consist of vertical, horizontal, and checkerboard patterns, as shown in Fig. 1.1. Because images are represented as linear combinations of these bases, the lack of bases aligned with other oblique directions hinders an efficient representation of directional image features, e.g., edges and lines, in such

orientations.

It is well recognized that the receptive fields of simple cells in mammalian visual cortex are band-pass and oriented [49, 102, 103, 149]. A more recent study [146] that evaluated sparse representations of natural images led to basis images closely matching the characteristics of the visual cells, supporting the hypothesis that the human visual system has been tuned to extract essential information in a natural scene with only a small number of active visual cells. These results also suggest that an efficient transform for image coding should provide a directional representation. To this end, several directional transforms that consist of directional bases other than the vertical and horizontal direction have been proposed in the literature.

1.2.1 Steerable Pyramid

One of the earlier works is the steerable pyramid [75]. It is an overcomplete transform with a redundancy factor of $5\frac{1}{3}$, i.e., generating $5\frac{1}{3}$ times as many coefficients as the image samples. The overcomplete representation contains bases at different resolution scales and aligned with four directions: 0°, 45°, 90° and 135°. An unique property of the steerable pyramid is that the coefficients corresponding to a basis rotated to an arbitrary orientation can be directly synthesized from the pyramid representation. This property enables image processing applications such as orientation and contour analysis [75], stereo matching and image enhancement [174], and orientation-invariant texture recognition [96]. However, because of the large amount of data expansion in the transform step, it is not suited for image coding.

1.2.2 Complex Wavelet Transform

The dual-tree complex wavelet transform (CWT) [113–116] is also an overcomplete representation, with a redundancy factor of 4 in two dimensions, that achieves nearly shift invariance and direction selectivity. A simplified version of the duel-tree CWT, referred to as the real oriented 2-D dual-tree transform [11], achieves only direction selectivity with a smaller redundancy factor of 2. Using the non-redundant DWT the energy in high-frequency patterns oriented at 45° and that oriented at 135° are

captured together in the HH subband, as suggested by the checkerboard pattern in the HH basis shown in Fig. 1.1-(b), whereas they are contained in separate subbands using the CWT or the simplified real transform. The dual-tree CWT has been applied to image denoising, geometry estimation and displacement estimation. Again, due to the redundancy in the transform domain, application to image coding has not been investigated. The mapping-based CWT later proposed in [70–72] constructs the CWT by first mapping the source onto a complex function space, followed by applying the DWT in the complex space. By controlling the redundancy in the mapping stage, a critically sampled and directional CWT can be obtained. However, image coding performance using the resulting non-redundant transform has not been studied.

1.2.3 Ridgelet Transform

Candès and Donoho proposed a continuous-space image representation named ridgelets [17, 20, 63, 64] that first maps a line singularity along an arbitrary orientation in the image into a point singularity with the Radon transform [50]. The point singularity in the Radon domain can then be effectively represented by the 1-D wavelet transform. The finite ridgelet transform (FRIT) [56, 58, 61] further extended the concept to work on discrete-space image samples by smartly combining the overcomplete finite Radon transform [12, 142] and the 1-D DWT to produce a critically sampled transform. Numerical results in [61] show that for a synthetic image containing a straight edge the FRIT outperforms the DWT in terms of nonlinear approximation, i.e., resulting in a smaller MSE in the reconstruction if only a given number of coefficients with the largest magnitude are kept and others set to zero in the inverse transform [134, 196]. Specifically, for this image when less than 1% of the largest coefficients are retained, the MSE from the FRIT is less than half of that from the DWT. However, no actual image coding result is provided.

1.2.4 Curvelet Transform

The curvelet transform [19] was also developed by Candès and Donoho initially in the continuous space. Incorporating ridgelets as a component step, the curvelet transform approximately partitions the frequency space in polar coordinates both radially and angularly, thus achieving directional selectivity. The discrete-space implementation of the curvelet transform [65, 178], an overcomplete representation, shows its effectiveness in image denoising. Further developments of the curvelet transform not using ridgelets are discussed both in the continuous space [21–23] and in the discrete space [18]. Nonlinear approximation performance for synthetic images containing smooth curves indicates that the curvelet transform is superior to DWT for this class of images.

1.2.5 Contourlet Transform

The directional filter bank proposed in [10] approximately partitions the frequency space into wedge-shaped directional subbands. The resulting transform is critically sampled and allows perfect reconstruction, both desirable properties for image coding. However, the low-frequency energy at the center of the frequency space is spread into the directional subbands, hampering energy compaction and making it less favorable for compression. The contourlet transform [57, 59, 60, 62] solves this problem by first constructing a Laplacian pyramid [15] and applying the directional filter bank to the high-pass images in the pyramid, resulting in a multi-resolution and multi-direction representation. However, the solution at the same time generates an overcomplete transform with a redundancy factor of $\frac{4}{3}$ from the Laplacian pyramid. Critically sampled versions of the contourlet transform have also been proposed [132, 144, 221]. The works in [132] and [144] do not include practical applications, whereas [221] reports better nonlinear approximation performance than the overcomplete contourlet and the DWT.

1.3 Direction-Adaptive Transforms

In [19], it has been demonstrated theoretically that for images containing smooth curves a non-adaptive directional representation can approach the nonlinear approximation performance of adaptive directional representations. However, in practice, non-adaptive directional transforms such as those discussed in Sec. 1.2 have not been able to produce rate-distortion image coding results that are generally better than non-directional transforms, e.g., the 2-D DCT and the 2-D DWT, primarily due to the redundant representation and lack of effective entropy coders. Direction-adaptive transforms turn out to be a superior alternative. A direction-adaptive transform typically consists of several modes, each mode providing directional bases dedicated to a certain orientation. According to the local directionality in the image, different modes are adaptively selected in different image regions at the encoder. Some schemes use forward adaptation such that the mode selections need to be signaled to the decoder as side information. For others using backward adaptation, these selections can be recovered at the decoder without any side information. As long as the transform in each mode is critically sampled, so is the overall adaptive transform. Additionally, a conventional non-directional transform can be incorporated as a non-directional mode, handling regions with no conspicuous directionality. Hence, with a proper design, the performance of an adaptive transform should never be inferior to the constituent non-directional transform.

1.3.1 Image Coding using Adaptive Transforms

The fundamental framework of image coding using adaptive transforms was introduced by Tasto and Wintz [180, 181]. An image is first divided into disjoint blocks, and the blocks are clustered into groups, each group coded with a transform and the associated quantization and entropy coding scheme that are adapted to the component blocks. Assuming a Gaussian source and given the number of groups and the size of each group, Tasto and Wintz presented an iterative algorithm that, at high rates, optimizes the clustering, the transforms and the associated rate allocation of the Gaussian coefficients. The rate overhead for the side information that signals

the mapping between the blocks and the groups, i.e., the mode selections, is taken into account in the algorithm. The resulting optimal transforms are the KLTs of the groups, and the KLT bases are inherently directional for groups containing directional image features.

Effros, Chou and Gray [66,67] generalized the framework as a code design problem for image coding systems, not constrained to Gaussian sources and the high-rate assumption, and additionally optimizing the size of each group. In their experiments, 64 modes, each specifying a KLT and the associated rate allocation, are trained from one image and then applied to code another image using forward adaptation, demonstrating a substantial gain over a JPEG-like non-adaptive approach. As described in [89–91], adaptive transforms based on the KLT can also be achieved with backward adaptation.

Despite their optimality, the KLTs generally do not have an efficient implementation. It may be more advantageous to design the modes with suboptimal transforms that can be easily constructed, such as the DCT and the DWT. One such example is the variable blocksize DCT that further divides a block into smaller partitions, which are square sub-blocks [33,189,190] or can be rectangular [54,55], and the DCT is separately applied within each partition. Different partitioning structures are selected for different blocks based on the local image content. Each structure corresponds to one mode of the overall adaptive transform, and the mode selections are signaled as side information. A rate-distortion optimized framework was proposed in [158] to select the best partitioning for every block, and the framework was also applied to locally adapt the decomposition structures of the DWT [159]. These adaptive transforms are, however, not directional.

1.3.2 DCT-based Direction-Adaptive Transforms

The work in [76, 218, 219] and our own approach in [27, 29] construct directional modes in direction-adaptive transforms by first performing one set of the 1-D DCTs, possibly with different lengths, along a certain orientation to directionally decorrelate the samples in an image block, followed by another supplementary set of the 1-D DCTs

to exploit the remaining correlation in other directions. A similar approach [217] uses a DCT-like non-directional mode and directional modes constructed with separable transforms customized from training data. Directional modes can also be obtained by permuting the block samples such that the permuted directional features are aligned either vertically or horizontally before applying a non-directional DCT-like transform [163]. Moreover, the lifting structure may be used for fast implementation of the DCT [126]. Using lifting, directional modes based on the DCT can be constructed, with directional bases extending across block boundaries [214], resulting in an adaptive transform that is no longer a block transform. The same technique can also be adopted to implement direction-adaptive lapped transforms [215]. All these adaptive transforms use forward adaptation.

1.3.3 DWT-based Direction-Adaptive Transforms

An early work of DWT-based direction-adaptive transforms is presented in [184]. An image is divided into blocks. Each block is then sheared through a reversible resampling filter such that directional features in the sheared block are oriented either vertically or horizontally. The 2-D DWT is applied to the sheared block and thus in effect provides directional bases via forward adaptation. The more recent work of the directionlets [191–194] constructs its directional modes by adapting both the wavelet filtering direction and the subsampling grid to the directionality in image blocks without resampling. The two approaches, among others [151, 200], share the same limitation, i.e., independent processing of image blocks that fails to exploit the correlation across block boundaries and produces blocking artifacts.

The bandelets representation [153, 154] does not have this limitation. The 2-D DWT is first applied to the image, followed by a bandeletization procedure that further removes the directional correlation in the LH, HL, and HH subbands using forward adaptation. The LL subband remains the same as in the 2-D DWT. Blocking artifacts are not observed since the block-wise operations in the bandeletization procedure are performed in the wavelet domain. However, because the procedure is essentially post-processing of the wavelet coefficients, the energy already contained in

the high-pass subbands can not be moved back to the low-pass subband, limiting its energy compaction ability.

The lifting structure enables locally adaptive wavelet filters together with a regular subsampling grid [179]. Using the lifting structure, several DWT-based approaches have been developed to locally adapt the filter coefficients [13,39,40,79], or the filtering directions [13, 80, 182], such that filtering is not performed across edges in images. These approaches use backward adaptation that eliminates the need for signaling the filter selections, i.e., the mode selections, to the decoder by assuming lossless compression [13] or knowledge of the quantization noise at the encoder [40, 79, 80], or constraining the selection process such that it can be reliably repeated at the decoder [182]. The gain of adaptation is limited due to these assumptions and constraints. No significant improvement on objective quality measurements over the 2-D DWT has been reported, although subjective improvement has been observed.

Other approaches that also adaptively select the filtering directions via lifting choose to explicitly signal the mode selections [52,53,129,200,201], i.e., using forward adaptation. Within this category, we have independently developed an approach that combines directional lifting with quincunx subsampling [30]. It was then extended to accommodate the case of conventional orthogonal subsampling, and to incorporate directional lifting and the bandeletization procedure into an unified framework [25]. More detailed analysis of the DWT-based direction-adaptive transforms using lifting is provided in our work in [26]. Thanks to the efficient representation of the filter selections, these approaches adapt to directional features more effectively and have demonstrated significant objective and subjective quality improvement for texture-rich images.

1.4 Summary of Contributions

This dissertation presents theoretical analysis and practical constructions of new direction-adaptive transforms for image coding. Some results have been published in [25–29]. The major contributions of the dissertation are summarized below:

- A novel statistical image model is proposed to analyze the performance of

transform coding. The model is able to represent locally varying textures in the image, hence suitable for the analysis of adaptive transforms. Using the model, we are able to theoretically quantify the potential improvements in image coding from using an adaptive transform instead of a fixed transform. The analysis shows that there is a substantial gain for images rich of edges and lines. For other images, significant improvements can still be expected locally around sharp directional features, rendered by directional bases of the adaptive transform. The model is also used to analyze the practical direction-adaptive transforms proposed in the dissertation.

- A new direction-adaptive partitioned block transform (DA-PBT) based on the DCT is proposed for coding of images and video sequences. At the same coding rate, the DA-PBT outperforms the conventional 2-D DCT by more than 2 dB for images with pronounced directional features. Since it avoids the typical ringing and checkerboard artifacts of the 2-D DCT, subjective improvements are even larger than indicated by the rate-distortion performance. The DA-PBT also outperforms a previously proposed directional DCT while demanding less computation. The combination of the DA-PBT with directional prediction within the image or interframe prediction for video is also explored, leading to further improvements in the performance of hybrid coding.

- A new direction-adaptive discrete wavelet transform (DA-DWT) is proposed for coding of still images. Using the lifting structure, the DA-DWT is able to locally adapt the filtering directions while ensuring the reversibility of the transform. In our experiments, the DA-DWT outperforms the conventional 2-D DWT by around 1 to 2 dB in PSNR for typical test images, and by up to 5.1 dB for special classes of images, consistent with the theoretical improvement derived for the DA-DWT using the image model. At low rates, different from the ringing and checkerboard artifacts typically observed in the 2-D DWT, the reconstruction error of the DA-DWT exhibits brushstroke-like artifacts that better preserve the geometric structure in the image, providing a visually more pleasing image representation.

1.5 Organization

The image model is presented in Chap. 2. In Sec. 2.1, we introduce the concept of modeling an image by a set of texture sources, each representing a region in the image. In Sec. 2.2, we propose a clustering algorithm that for a given image optimizes the image regions and the parameters characterizing the corresponding texture sources. Performance analysis based on the model is presented in Sec. 2.3.

The DA-PBT is introduced in Chap. 3. We first describe different modes of the DA-PBT in Sec. 3.1. The application of the DA-PBT to image coding in practice is discussed in Sec. 3.2. The proposed transform can be combined with block-based prediction to improve the performance of hybrid coding, as explained in Sec. 3.3. Theoretical analysis based on the image model is included in Sec. 3.4. Experimental results demonstrating the superior performance of the proposed transform are shown in Sec. 3.5.

The DA-DWT is discussed in Chap. 4. In Sec. 4.1, we describe how different modes of the DA-DWT are constructed using directional lifting. A practical image coding framework using the DA-DWT is discussed in Sec. 4.2. Theoretical analysis of the DA-DWT and performance evaluation using the image model are presented in Sec. 4.3. Finally, experimental results demonstrating the superior performance of the DA-DWT both objectively and subjectively are included in Sec. 4.4.

Chapter 2

Statistical Image Model for Transform Coding Analysis

A model that captures important attributes of a signal is an essential tool to analyze the performance of signal processing algorithms. In this chapter, we propose a statistical image model to study the performance of transform coding of images. In particular, using the image model we quantify the theoretical performance improvement from incorporating transforms locally adapted to image content instead of applying a fixed transform, further providing insights to the design of practical adaptive transforms for image coding. In the subsequent chapters in this dissertation, the model will also be used to analyze the performance of the proposed direction-adaptive transforms.

In Sec. 2.1, we introduce the concept of modeling an image by a mixture of texture sources. Each texture source is a 2-D stationary random field characterized by a parametric autocorrelation function, modeling a particular texture that appears in the image. An image is segmented into regions containing different textures, each represented by a texture source. In Sec. 2.2, we propose a clustering algorithm that for a given image optimizes the segmentation and the parameters characterizing the corresponding texture sources. Performance analyses based on the model are presented in Sec. 2.3.

2.1 Image Modeling with Texture Sources

2.1.1 Mixture Model

Stochastic image models have long been adopted to represent images for theoretical analysis [104, 106]. These models often provide a global description of an image using a single set of statistics. However, different regions in an image may have very different characteristics. For instance, in the *Monarch* image shown in Fig. 2.1, the characteristics of the high-contrast patterns in the foreground clearly differ from those of the blurred background. Moreover, the various shapes and orientations in the foreground patterns also lead to distinct characteristics that cannot be easily represented by a single set of statistics. To analyze locally adaptive algorithms, a suitable model should be able to reflect these local variations.

As a better alternative, a mixture model represents an image as a mixture of multiple components, each described by a set of statistics reflecting certain local characteristics. The most widely adopted mixture model for images is the Gauss mixture model where each component is a Gaussian source. The model is typically constructed by the expectation-maximization (EM) algorithm or its variations using the maximum-likelihood criterion [51, 73], assuming that each block in the image is in fact a sample of the Gauss mixture. Applications of the Gauss mixture model constructed using the EM algorithm include, for instance, 2-D hidden Markov model (HMM) for image classification [125], image segmentation and image query [24], nonlinear prediction [220] and feature selection [119].

As shown in [4, 92, 95], the Gauss mixture model can also be constructed by the Lloyd algorithm [127, 130]. Unlike the EM algorithm, the Lloyd algorithm does not assume that the image blocks are samples of the Gauss mixture and only finds a set of Gaussian components and the mapping between the blocks and the components that minimizes the overall distortion measure. Using quantization mismatch (QM) or minimum discrimination information (MDI) as the distortion measure [4, 92, 95], the resulting Gauss mixture model can be used to design robust code for image compression and has also been applied to applications in content classification [4]. Note that

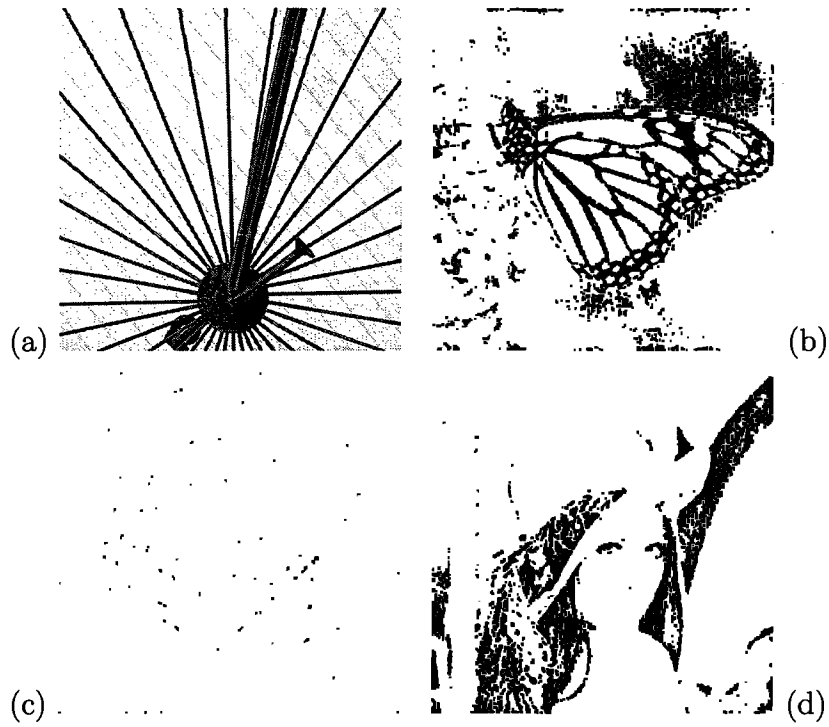


Figure 2.1: 512 × 512 8-bit grayscale images (a) *Spoke*, (b) *Monarch*, (c) *Pentagon* and (d) *Lena*.

in the case of the MDI distortion measure, the components are 2-D recursible autoregressive sources [123], analogous to the 1-D case successfully used in linear predictive coded (LPC) speech compression systems [93, 94].

2.1.2 Texture Source

To study the performance of locally adaptive algorithms, in particular, adaptive transforms for image coding, we propose to model an image by a mixture of texture sources. Each texture source is itself a stationary random field representing a particular texture that appears in the image. Any local image neighborhood is associated with a texture source best describing the local statistics. Therefore, to model the performance of an algorithm for the image, the performance in a local neighborhood can be estimated from the analytical performance for the associated texture source. Additionally, the image-wise performance can also be derived by averaging over all neighborhoods.

A texture source is first defined as a 2-D continuous-space stationary zero-mean unit-variance random field, denoted by $\tilde{t}(\mathbf{p}) = \tilde{t}(p_x, p_y)$, $\mathbf{p} = (p_x, p_y)^T \in \mathbb{R}^2$, modeling the corresponding texture in the scene. The scene is then converted by an imaging device into a discrete-space texture source, consisting of samples of a filtered and scaled version of the continuous-space texture source. The filtering operation results from the aperture function of the imaging device that mainly accounts for the anti-aliasing process, and the scaling normalizes the discrete-space random field so that it is also unit-variance. Consequently, the discrete-space texture source is a 2-D discrete-space stationary zero-mean unit-variance random field, denoted by $t[\mathbf{l}] = t[l_x, l_y]$, $\mathbf{l} = (l_x, l_y)^T \in \mathbb{Z}^2$. Using the discrete-space texture sources, any local image neighborhood is modeled as a 2-D segment of a realization of a texture source, scaled in intensity to match the actual local variance in the neighborhood and shifted to match the local mean.

Each texture source, either continuous-space or discrete-space, is characterized by its power spectral density (PSD) or equivalently the autocorrelation function. It is desirable to have a simple parametric representation of the PSD and the autocorrelation to facilitate the modeling process as well as the subsequent analysis. Despite the desired simplicity, the parametrization should be able to describe a variety of textures contained in typically images. To this end, we define two classes of image textures: the regular textures, such as edges, lines (double edges), and stripes (periodic edges), and the irregular textures, accounting for other, usually more complex, features. A texture source consists of two components representing the two classes of textures. In the following sections, we first describe a parametric formulation to model the class of the irregular textures in Sec. 2.1.3, followed by the parameterization for the regular textures in Sec. 2.1.4, both defined in the continuous space. Finally, the composition of the two components and the conversion from the continuous space to the discrete space are discussed in Sec. 2.1.5.

2.1.3 Elliptic Random Field

In this section, we first consider the continuous-space irregular textures. These zero-mean textures contain complex structures so that in any 1-D neighborhood zero-crossings occur in a random manner. This zero-crossing pattern is similar to that of a mean-removed random telegraph signal, a bi-level signal with the changes in level (zero-crossing) occurring according to a Poisson process [121]. It can be shown that the autocorrelation function, denoted by $r(\tau)$, of a zero-mean unit-variance random telegraph signal can be expressed as $r(\tau) = \exp(-2\lambda_p|\tau|)$ where λ_p is the rate of the underlying Poisson process. Due to this similarity, we assume that the correlation in the irregular textures generally decays exponentially with the distance between two points, also consistent with the exponentially decaying function in both the isotropic model [147, 167] and the separable model [98] widely adopted in the literature.

Denote the autocorrelation function of a 2-D continuous-space random field by $r_{\bar{t}\bar{t}}(\boldsymbol{\tau}) = r_{\bar{t}\bar{t}}(\tau_x, \tau_y)$, $\boldsymbol{\tau} = (\tau_x, \tau_y)^T \in \mathbb{R}^2$. In the isotropic model, assuming unit variance,

$$r^{iso}_{\bar{t}\bar{t}}(\boldsymbol{\tau}|\lambda_r) = \exp(-\lambda_r(\tau_x^2 + \tau_y^2)^{\frac{1}{2}}), \tag{2.1}$$

and in the separable model

$$r^{sep}_{\bar{t}\bar{t}}(\boldsymbol{\tau}|\lambda_x, \lambda_y) = \exp(-\lambda_x|\tau_x|) \cdot \exp(-\lambda_y|\tau_y|). \tag{2.2}$$

The contours of equal autocorrelation are circles in the isotropic model, whereas they are rhombi in the separable model, with the diagonals aligned with the vertical and the horizontal axes. Consequently, the former is not able to represent any directionality in image textures, and the latter can only describe directional textures oriented either vertically or horizontally. To better capture the directionality in the irregular textures, we extent the isotropic model and propose the following unit-variance autocorrelation

function:

$$r^I_{\tilde{t}\tilde{t}}(\boldsymbol{\tau}|\theta,\lambda_a,\lambda_b) = \exp(-(\lambda_a^2\tau_a^2 + \lambda_b^2\tau_b^2)^{\frac{1}{2}}), \tag{2.3a}$$

$$\begin{pmatrix} \tau_a \\ \tau_b \end{pmatrix} = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \boldsymbol{\tau}, \tag{2.3b}$$

where $0 \leq \theta < \pi$ and $0 < \lambda_b \leq \lambda_a$. The autocorrelation decays exponentially by $\exp(-\lambda_a|\tau_a|)$ along θ, and it decays at a smaller or equal rate by $\exp(-\lambda_b|\tau_b|)$ along the direction orthogonal to θ. The angle θ therefore indicates the direction of the largest variation in the texture. The autocorrelation along other directions also decays exponentially at different rates such that the contours of equal autocorrelation are concentric ellipses where the minor axis is aligned with θ. Additionally, the length of the minor axis and that of the major axis are proportional to λ_a^{-1} and λ_b^{-1} respectively. We refer to a random field characterized by the autocorrelation function in (2.3) as an elliptic random field, from the structure of the autocorrelation function. The elliptic random field also generalizes the isotropic model in (2.1) since $r^{iso}_{\tilde{t}\tilde{t}}(\boldsymbol{\tau}|\lambda_r) = r^I_{\tilde{t}\tilde{t}}(\boldsymbol{\tau}|0,\lambda_r,\lambda_r)$.

Some example realizations of elliptic random fields, i.e., continuous-space irregular textures, are shown in Fig. 2.2. These examples are generated with $\lambda_a = -\ln(0.8)f_s$ where f_s denotes both the horizontal and the vertical sampling frequency, $\lambda_b = \lambda_a/k_b$ with $k_b = 2$, 8 and 32, and $\theta = 0°$, 22.5°, 45°, 67.5° and 90°, corresponding to textures with different degrees of directionality and different orientations. The value $\lambda_a = -\ln(0.8)f_s$ is chosen in these examples such that along θ the correlation between two points separated by $\tau_a = f_s^{-1}$, i.e., the sampling period, equals 0.8. Recall that a texture source models a local image neighborhood normalized by the local mean and variance. From our observation a correlation of 0.8 between two neighboring pixels is typical when measured in such normalized neighborhoods. This value is smaller than the correlation around 0.95 often reported, for a similar set of test images, from measurements averaging over the entire image, normalized by the global mean and variance [98, 147, 167]. This is because, qualitatively speaking, a seemingly smooth region observed at the image-wise scale can appear noisy, hence less correlated, when

being scrutinized at a local scale.

Denote the PSD of a 2-D continuous-space random field by $\Phi_{\tilde{ii}}(\boldsymbol{\omega}) = \Phi_{\tilde{ii}}(\omega_x, \omega_y)$, $\boldsymbol{\omega} = (\omega_x, \omega_y)^T \in \mathbb{R}^2$. From (2.3), the PSD of an elliptic random field can be expressed as

$$\Phi^I_{\tilde{ii}}(\boldsymbol{\omega}|\theta, \lambda_a, \lambda_b) = \frac{2\pi}{\lambda_a \lambda_b}(1 + (\frac{\omega_a}{\lambda_a})^2 + (\frac{\omega_b}{\lambda_b})^2)^{-\frac{3}{2}}, \tag{2.4a}$$

$$\begin{pmatrix} \omega_a \\ \omega_b \end{pmatrix} = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \boldsymbol{\omega}. \tag{2.4b}$$

For an irregular texture modeled as an elliptic random field, the contours of equal PSD are also concentric ellipses with the major axis aligned with θ. The length of the major axis and that of the minor axis are proportional to λ_a and λ_b respectively.

2.1.4 Periodic-Wave Random Field

In Sec. 2.1.3, for irregular textures we assume that in any 1-D neighborhood zero-crossings occur in a random manner, so that the correlation decays exponentially with the distance between two points. For regular textures containing essentially lines and edges, however, this assumption does not hold.

Consider a line in a continuous-space zero-mean regular texture and a group of 1-D neighborhoods that are orthogonal to the line and intersect the line. In this example, there are always two zero-crossings in every neighborhood, one at each edge of the line, with the same amount of separation. This fixed zero-crossing pattern can no longer be approximated by a Poisson random process. Instead, we can approximate these neighborhoods by segments in a 1-D random square-wave signal, defined as

$$\tilde{s}(p|T_s, r_s) = \begin{cases} A_0, & \exists\, k \in \mathbb{Z} \text{ s.t. } 0 \le p + kT_s + \xi < r_s T_s; \\ A_1, & \text{otherwise,} \end{cases} \tag{2.5}$$

where T_s is the period, A_0 and A_1 are two intensity levels, $0 < r_s < 1$ determines the duration of the two levels, and ξ is a random phase. For the above example, any 1-D neighborhood intersecting with the line can be modeled as a segment of a random

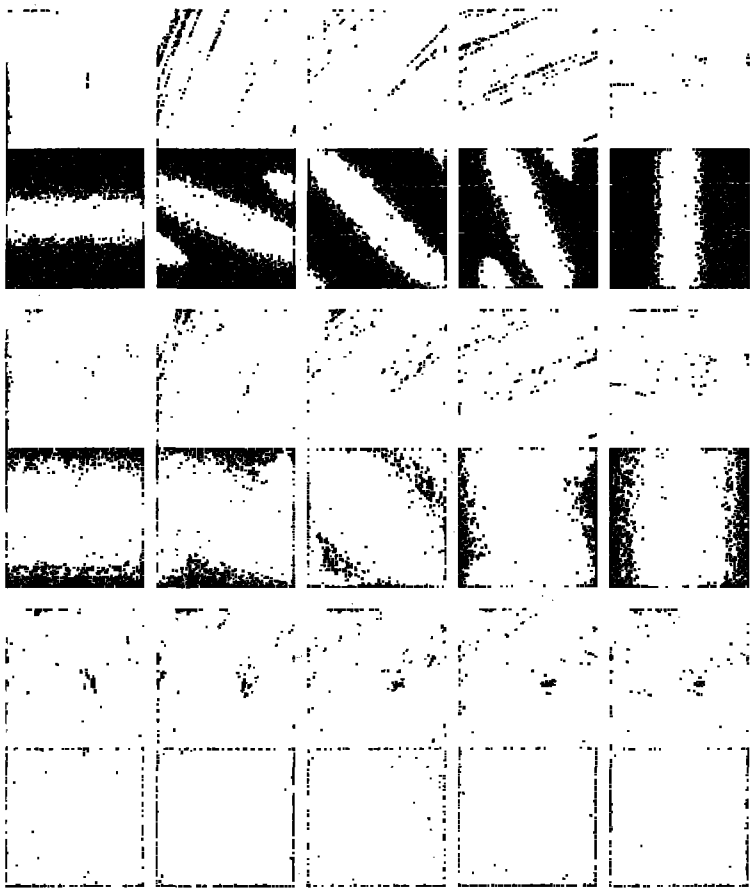


Figure 2.2: Elliptic random field examples: $\lambda_a = -\ln(0.8) f_s$, $\lambda_b = \lambda_a / k_b$ where $k_b = 2$, 8 and 32 from bottom to top, and $\theta = 0°$, 22.5°, 45°, 67.5° and 90° from left to right. For each example, the top plot shows a realization of the continuous-space random field, i.e., the irregular texture being modeled, and the bottom plot shows the discrete-space PSD in logarithmic scale where the square region corresponds to $-\pi \leq \Omega_x \leq \pi$ from left to right and $-\pi \leq \Omega_y \leq \pi$ from bottom to top.

square-wave signal with T_s close to the neighborhood size and r_s selected according to the width of the line. Similarly, a neighborhood containing only a single edge, i.e., a step function, is modeled as a segment of a random square-wave signal with T_s close to twice the neighborhood size and $r_s = 0.5$ so that there is always one edge in every segment. Modeling a neighborhood containing periodic stripes can be done in the same fashion. Finally, it is straightforward to show that the autocorrelation function of a zero-mean unit-variance random square-wave signal also has a periodic-wave

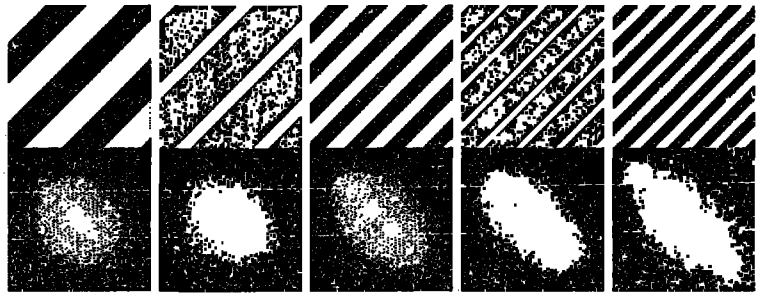


Figure 2.3: Periodic-wave random field examples: $K = 3$, $\theta = 45°$, and from left to right (ω_s, r_s) equals $(0.1\pi f_s, 0.5)$, $(0.1\pi f_s, 0.2)$, $(0.2\pi f_s, 0.5)$, $(0.2\pi f_s, 0.2)$ and $(0.3\pi f_s, 0.5)$. For each example, the top plot shows a realization of the continuous-space random field, i.e., the regular texture being modeled, and the bottom plot shows the discrete-space PSD in logarithmic scale.

structure, more specifically a periodic sequence of mean-removed triangles, expressed by

$$r_{\tilde{s}\tilde{s}}(\tau|T_s, r_s) = \frac{2r_s}{1-r_s} \sum_{k=1}^{\infty} \text{sinc}(kr_s)^2 \cos(k\frac{2\pi}{T_s}\tau). \tag{2.6}$$

Note that $r_{\tilde{s}\tilde{s}}(\tau|T_s, r_s) = r_{\tilde{s}\tilde{s}}(\tau|T_s, 1-r_s)$.

Based on the above discussion, we propose the following unit-variance autocorrelation function, in the same form as (2.6), to characterize regular textures:

$$r^R_{\tilde{t}\tilde{t}}(\boldsymbol{\tau}|\theta, \omega_s, r_s) = \sum_{k=1}^{K} \gamma(k, r_s) \cos(k\omega_s \tau_a), \tag{2.7a}$$

$$\gamma(k, r_s) = \frac{\text{sinc}(kr_s)^2}{\sum_{k=1}^{K} \text{sinc}(kr_s)^2}, \tag{2.7b}$$

where τ_a is defined in (2.3b), $0 \leq \theta < \pi$, $\omega_s > 0$ and $0 < r_s \leq \frac{1}{2}$. Similar to the elliptic random fields, the tangent of the edges and lines in a regular texture aligns with θ. Along θ, the autocorrelation function is essentially the same as (2.6) with $\omega_s = \frac{2\pi}{T_s}$ and the infinite sum approximated by the sum of the first K harmonics. In practice, we use $K = 3$. Orthogonal to θ the autocorrelation is constant since the intensity does not change along edges. We refer to a random field characterized

by the autocorrelation function in (2.7) as a periodic-wave random field, again from the structure of the autocorrelation function. Note that the periodic-wave random fields enable modeling of negative autocorrelation values, often observed in actual measurements around edges, which cannot be described by the elliptic random fields. Some example realizations of the periodic-wave random fields, i.e., the continuous-space regular textures, are shown in Fig. 2.3.

The PSD of a periodic-wave random field can be expressed as

$$\Phi^R_{\bar{t}\bar{t}}(\boldsymbol{\omega}|\theta,\omega_s,r_s)=\sum_{k=1}^{K}\gamma(k,r_s)\pi(\delta(\omega_a-k\omega_s)\delta(\omega_b)+\delta(\omega_a+k\omega_s)\delta(\omega_b)), \tag{2.8}$$

where ω_a and ω_b are defined in (2.4b). A regular texture appears as pairs of 2-D delta impulses along θ in the PSD, accounting for the off-center peaks frequently observed in spectrum estimation from discrete-space samples [148]. In practice, to facilitate numerical evaluation in the subsequent analysis, we approximate $\delta(\omega_a \pm k\omega_s)\delta(\omega_b)$ in (2.8) by $\delta(\omega_a \pm k\omega_s)\delta(\omega_b) * \frac{1}{2\pi}\Phi^I_{\bar{t}\bar{t}}(\boldsymbol{\omega}|0,\lambda_r,\lambda_r)$, where $\Phi^I_{\bar{t}\bar{t}}$ is defined in (2.4), the symbol $*$ denotes the 2-D convolution operation, and $\lambda_r = 10^{-3}\pi f_s$, i.e., the PSD of the isotropic model in (2.1) with a small λ_r, scaled by $\frac{1}{2\pi}$ and shifted by $\pm k\omega_s$ along θ. Correspondingly, the autocorrelation function in (2.7) is multiplied by $r^I_{\bar{t}\bar{t}}(\boldsymbol{\tau}|0,\lambda_r,\lambda_r)$ defined in (2.3).

2.1.5 Discrete-Space Texture Source

To account for textures containing both an irregular component and a regular component, we define a continuous-space texture source as a 2-D stationary zero-mean unit-variance random field combining an elliptic random field with a periodic-wave random field:

$$r_{\bar{t}\bar{t}}(\boldsymbol{\tau}|\Theta)=\beta\cdot r^I_{\bar{t}\bar{t}}(\boldsymbol{\tau}|\theta,\lambda_a,\lambda_b)+(1-\beta)\cdot r^R_{\bar{t}\bar{t}}(\boldsymbol{\tau}|\theta,\omega_s,r_s), \tag{2.9a}$$

$$\Phi_{\bar{t}\bar{t}}(\boldsymbol{\omega}|\Theta)=\beta\cdot \Phi^I_{\bar{t}\bar{t}}(\boldsymbol{\omega}|\theta,\lambda_a,\lambda_b)+(1-\beta)\cdot \Phi^R_{\bar{t}\bar{t}}(\boldsymbol{\omega}|\theta,\omega_s,r_s), \tag{2.9b}$$

where $\Theta = \{\beta, \theta, \lambda_a, \lambda_b, \omega_s, r_s\}$ and $0 \leq \beta \leq 1$. The two components share the same θ for simplicity, and there are 6 parameters in total to describe a texture source. Note that taking a weighted average of the two components by β explicitly assumes that they are uncorrelated.

To derive the discrete-space texture source from the continuous-space definition, we first consider the aperture function of the imaging device, denoted by $\tilde{h}_a(\mathbf{p})$, and the corresponding frequency response, denoted by $\tilde{H}_a(\boldsymbol{\omega})$. The aperture function accounts for all filtering operations in the device, and the ideal aperture function is the ideal anti-aliasing filter with a cut-off frequency at $\omega_x = \pm\pi f_s$ and $\omega_y = \pm\pi f_s$, where f_s denotes the sampling frequency in both dimensions. For practical imaging devices, we model the aperture function as a separable filter for simplicity, i.e., $\tilde{H}_a(\boldsymbol{\omega}) = \tilde{H}_{a,1}(\omega_x|f_s)\tilde{H}_{a,1}(\omega_y|f_s)$. The 1-D filter $\tilde{H}_{a,1}(\omega|f_s)$ is modeled as a root-raised-cosine filter with the roll-off factor $\beta_r = 0.5$ such that $|\tilde{H}_{a,1}(\omega|f_s)| = 0$ for $|\omega| > 1.5\pi f_s$.

Given $r_{\tilde{t}\tilde{t}}(\boldsymbol{\tau}|\Theta)$ and $\Phi_{\tilde{t}\tilde{t}}(\boldsymbol{\omega}|\Theta)$ in (2.9), the autocorrelation function and the PSD of the anti-aliased source, denoted by $r_{\hat{t}\hat{t}}(\boldsymbol{\tau}|\Theta)$ and $\Phi_{\hat{t}\hat{t}}(\boldsymbol{\omega}|\Theta)$ respectively, are obtained by

$$r_{\hat{t}\hat{t}}(\boldsymbol{\tau}|\Theta) = r_{\tilde{t}\tilde{t}}(\boldsymbol{\tau}|\Theta) * \tilde{h}_a(\boldsymbol{\tau}) * \tilde{h}_a(-\boldsymbol{\tau}), \tag{2.10a}$$

$$\Phi_{\hat{t}\hat{t}}(\boldsymbol{\omega}|\Theta) = \Phi_{\tilde{t}\tilde{t}}(\boldsymbol{\omega}|\Theta) \cdot |\tilde{H}_a(\boldsymbol{\omega})|^2. \tag{2.10b}$$

Note that $\Phi_{\hat{t}\hat{t}}(\boldsymbol{\omega}|\Theta)$ is now band-limited since $|\tilde{H}_a(\boldsymbol{\omega})| = 0$ for $|\omega_x|$ or $|\omega_y|$ larger than $1.5\pi f_s$. Finally, the autocorrelation function of the discrete-space unit-variance texture source is obtained by

$$r_{tt}[\mathbf{d}|\Theta] = r_{tt}[d_x, d_y|\Theta] = \frac{1}{r_{\hat{t}\hat{t}}(0,0|\Theta)} r_{\hat{t}\hat{t}}(\frac{d_x}{f_s}, \frac{d_y}{f_s}|\Theta), \tag{2.11}$$

and the corresponding discrete-space PSD is

$$\begin{aligned}\Phi_{tt}(e^{j\boldsymbol{\Omega}}|\Theta) &= \Phi_{tt}(e^{j\Omega_x}, e^{j\Omega_y}|\Theta) \\ &= \frac{1}{r_{\tilde{t}\tilde{t}}(0,0|\Theta)} \sum_{z_x=-1}^{1} \sum_{z_y=-1}^{1} \Phi_{\hat{t}\hat{t}}((\Omega_x - 2\pi z_x)f_s, (\Omega_y - 2\pi z_y)f_s|\Theta).\end{aligned} \tag{2.12}$$

Some examples of the discrete-space PSD with $\beta = 1$ in (2.9b), i.e., the elliptic random fields, are shown in Fig. 2.2, and others with $\beta = 0$, i.e., the periodic-wave random fields, are shown in Fig. 2.3. Notice that high-frequency patterns in the continuous-space image textures may lead to aliasing components in the discrete-space image samples due to the non-ideal anti-aliasing filter. These aliasing effects often observed in actual images can be captured by the proposed model as shown in Fig. 2.2 ($k_b = 32$, $\theta = 22.5°$, $67.5°$).

2.2 Texture Source Clustering

Having defined the texture sources in Sec. 2.1, two questions remain to be answered: how to divide a given image into regions of different textures, and what are the texture sources best describing these regions? The answers to the two questions depend on each other, as we shall see in the following discussion.

2.2.1 Image Neighborhood Modeling

Recall that, in the proposed image model, any local image neighborhood is modeled as a 2-D segment of a realization of a particular discrete-space zero-mean unit-variance texture source, scaled to match the local variance and shifted to match the local mean. We do not explicitly impose any constraint on the size or the shape of the neighborhood. However, it is implied that a neighborhood contains homogeneous content so that it can be well represented by a single texture source.

To represent an image neighborhood with a texture source, we divide the image into L_B disjoint blocks of $S_B \times S_B$ pixels. Each block is associated with a local mean and a local variance, estimated from the image pixels in the vicinity of the block and denoted by μ_b and σ_b^2 respectively. Each block is also associated with a texture source, and any image neighborhood centered in the block is then modeled by this texture source, after normalization by σ_b and μ_b. In general, the blocksize S_B should be kept small to reflect the locally varying statistics in the image. However, as an example, for a 512×512 image and 8×8 blocks, there are 4096 blocks which requires

the same number of texture sources to represent, making the model impractical.

To further simplify the model, we consider the case that only L_C texture sources, $L_C \ll L_B$, are included to model an image. The L_B blocks are grouped into L_C clusters. Each cluster contains blocks located in image regions with similar textures and is represented by a single texture source. A clustering algorithm, similar to the Lloyd algorithm [127], is proposed to optimize both the choice of the L_C texture sources and the assignments that associate each block to one of the texture sources, as described in the following sections.

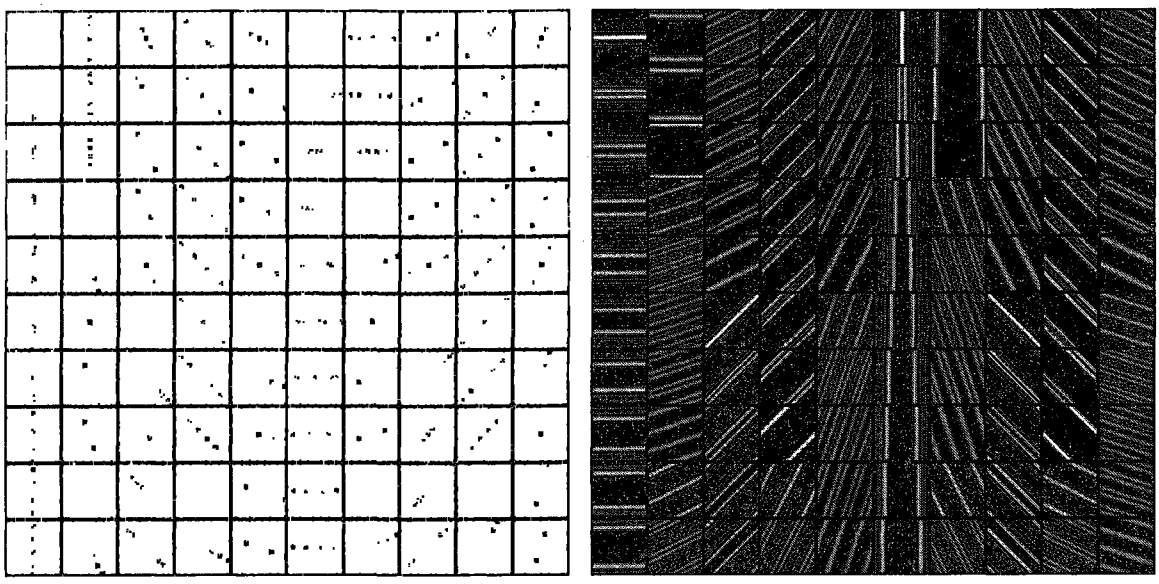
2.2.2 Cost Function

In this section we first define the cost function for the clustering algorithm, analogous to the distortion measure in the Lloyd algorithm [127]. For an image block, the cost between the block and a texture source reflects the modeling error from representing the image neighborhoods centered in the block by the texture source.

The main purpose of the image model is to analyze the rate-distortion performance of transform coding of images in a general framework. Specifically, later explained in Sec. 2.3.2, coding of the coefficients in a subband in a local neighborhood is assumed to achieve the rate-distortion function of a Gaussian memoryless source, i.e.,

$$R_s(\lambda) = \max\{0, \frac{1}{2}\log_2 \frac{\sigma_s^2}{\lambda}\}, \quad D_s(\lambda) = \min\{\lambda, \sigma_s^2\}, \tag{2.13}$$

where R_s and D_s are the rate in bits per coefficient and the MSE distortion for coding subband s respectively, σ_s^2 denotes the variance of the subband and λ controls the rate-distortion trade-off [44]. Each subband can be considered as a filtered version of the image where the filter is determined by the corresponding basis of the transform. To model accurately the local rate-distortion performance of a particular subband around a block, the actual variance of the filter output, hence the subband variance, measured in the block should be closely approximated by the variance derived from the associated texture source. Consequently, for any filter corresponding to a basis of a transform of interest, a well-designed cost function should possess the property that, for every block, the output variance derived from a texture source with a smaller



(a) impulse response (b) magnitude frequency response

Figure 2.4: The set of 100 filters used in the cost function of the clustering algorithm. In (a), gray represents magnitude 0. In (b), each square region corresponds to the 2-D discrete-space frequency $-\pi \leq \Omega_x \leq \pi$ from left to right and $-\pi \leq \Omega_y \leq \pi$ from bottom to top, and black represents magnitude 0.

cost is in general closer to the measured variance.

To come up with a cost function with this property, we consider the set of 100 filters with their impulse responses and the corresponding frequency responses shown in Fig. 2.4-(a) and Fig. 2.4-(b) respectively. These filters correspond to 12 subsets of the 1-D DCT bases along different directions in a maximum support of 13×13 pixels. For instance, the first subset contains the bases of the 13-point DCT aligned vertically, shown in the first column and the beginning of the second column in Fig. 2.4-(a). The second subset contains the bases of the 5-point DCT along a direction with a slope $\frac{\triangle y}{\triangle x} = -3$, shown in the second column in Fig. 2.4-(a). The set of filters is selected for its simplicity and its ability to localize spectral components covering the entire 2-D frequency space with different orientations, as shown in Fig. 2.4-(b). We apply each of these filters to the image, and denote the measured output variance of the i-th filter in the b-th block by $\bar{\sigma}^2_{b,i}$. Similarly, denote the output variance of the i-th filter derived from a texture source described by a parameter set Θ by $\sigma^2_{i|\Theta}$. We argue that

a texture source with $\sigma^2_{i|\Theta}$ closely approximating $\bar{\sigma}^2_{b,i}/\sigma^2_b$ (σ^2_b defined in Sec. 2.2.1), for all $i = 0, \cdots, 99$, captures the spectral structure in the vicinity of the b-th block, and therefore it is able to provide an accurate model for the output variances of other filters, e.g., the subband variances resulting from a transform. Based on this argument, for the b-th block, we define the cost for associating the block with a texture source described by Θ as

$$d_b(\Theta) = \sum_{i=0}^{99} (\bar{\sigma}^2_{b,i})^{-k} \cdot (\sigma^2_b \sigma^2_{i|\Theta} - \bar{\sigma}^2_{b,i})^2, \tag{2.14}$$

where $0 \leq k \leq 2$. Before discussing the choice of k, we first note that in (2.13), for a small λ where $\lambda \leq \sigma^2_s$ for subband s, the distortion equals λ and the modeling error in rate from using a model variance $\hat{\sigma}^2_s$ instead of the actual σ^2_s is proportional to $\log_2(\hat{\sigma}^2_s/\sigma^2_s)$. For a large λ where $\lambda > \sigma^2_s$, the rate is zero and the modeling error in distortion is $\hat{\sigma}^2_s - \sigma^2_s$.

In (2.14), when $k = 2$ the cost is the sum of the squared normalized errors, i.e., $(\sigma^2_b \sigma^2_{i|\Theta})/\bar{\sigma}^2_{b,i} - 1$. In this case, a small cost indicates that the normalized errors for all filters in the set are small in magnitude, and thus presumably the normalized error of the subband variance resulting from any transform basis is also small. This ensures that, with a small λ, both the distortion and the rate in (2.13) for a subband are accurately modeled. However, for a large subband variance, a small normalized error can lead to a large non-normalized error, therefore for a large λ resulting in a significant modeling error in distortion. On the contrary, when $k = 0$ the cost is the sum of the squared non-normalized errors, i.e., $\sigma^2_b \sigma^2_{i|\Theta} - \bar{\sigma}^2_{b,i}$. With a large λ, a small cost, hence small non-normalized errors, gives an accurate model for both the rate and the distortion. However, for a small variance, a small non-normalized error can still lead to a large normalized error, resulting in a substantial modeling error in rate with a small λ. For most values of λ corresponding to the typical rate-distortion operating points for image coding in practice, λ is larger than the subband variance for some subbands and smaller for the others. As a compromise, we choose $k = 1.5$ from experiments such that the model rate-distortion performance best matches the actual performance for several transforms we considered.

To improve the accuracy of the proposed image model, we further include an additive white-noise component associated with each block such that the image neighborhoods centered in the block, after normalization by the local mean and variance, is modeled by a random field characterized by the autocorrelation function $\alpha(b)\delta[0,0]+(1-\alpha(b))r_{tt}[\mathbf{d}|\Theta]$ where $0 \leq \alpha(b) \leq 1$ determines the relative white-noise level associated with the b-th block, and $r_{tt}[\mathbf{d}|\Theta]$ denotes the discrete-space autocorrelation function defined in (2.11). With the white-noise component, the cost function in (2.14) is modified accordingly to

$$d_b(\Theta, \alpha(b)) = \sum_{i=0}^{99} (\bar{\sigma}_{b,i}^2)^{-k} \cdot (\sigma_b^2(\alpha(b) + (1-\alpha(b))\sigma_{i|\Theta}^2) - \bar{\sigma}_{b,i}^2)^2. \tag{2.15}$$

2.2.3 Clustering Algorithm

Let Θ_c denote the set of 6 parameters characterizing the c-th texture source, and $m(b) = c$ denote the membership function that associates the b-th block with the c-th texture source. We now describe a clustering algorithm that optimizes the texture sources Θ_c, $c = 0, \cdots, L_C-1$, and the relative white-noise levels $\alpha(b)$ and the membership function $m(b)$, $b = 0, \cdots, L_B-1$, such that the overall cost $\sum_{b=0}^{L_B-1} d_b(\Theta_{m(b)}, \alpha(b))$ is minimized. We begin the algorithm by setting $\alpha(b) = 0$ and initializing $m(b)$ via evenly partitioning the blocks into L_C clusters, followed by the three optimization steps described below.

In Step 1, given the initialized $\alpha(b)$ and $m(b)$, for each texture source the parameters are optimized to minimize the total cost in the corresponding cluster, i.e.,

$$\Theta_c = \underset{\Theta}{\operatorname{argmin}} \sum_{\{b|m(b)=c\}} d_b(\Theta, \alpha(b)), \tag{2.16}$$

hence also minimizing the overall cost. The procedure that optimizes the parameters will be further discussed in Sec. 2.2.4.

In Step 2, given the texture sources resulting from Step 1 and the initialized $m(b)$, for every block the white-noise level $\alpha(b)$ minimizing the cost $d_b(\Theta_{m(b)}, \alpha(b))$ is solved

analytically by setting the derivative of (2.15) with respect to $\alpha(b)$ to zero, i.e.,

$$\alpha(b) = \max\{0, \min\{1, \frac{\sum_{i=0}^{99}(\bar{\sigma}_{b,i}^2)^{-k}(\bar{\sigma}_{b,i}^2 - \sigma_b^2\sigma_{i|\Theta_{m(b)}}^2)(1-\sigma_{i|\Theta_{m(b)}}^2)}{\sigma_b^2\sum_{i=0}^{99}(\bar{\sigma}_{b,i}^2)^{-k}(1-\sigma_{i|\Theta_{m(b)}}^2)^2}\}\}, \tag{2.17}$$

for all $b = 0, \cdots, L_B - 1$, again minimizing the overall cost.

In Step 3, given the texture sources and $\alpha(b)$, $m(b)$ is re-selected by

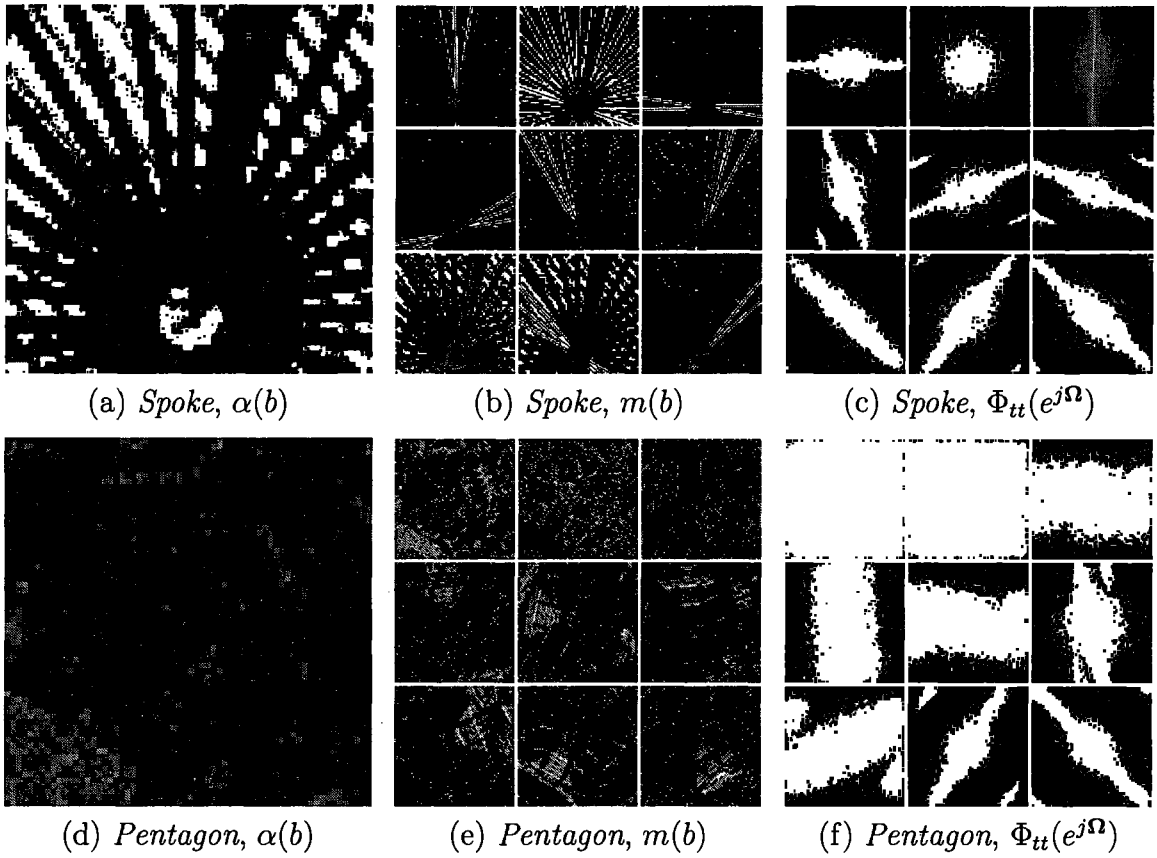
$$m(b) = \underset{c=0,\cdots,L_C-1}{\operatorname{argmin}} d_b(\Theta_c, \alpha(b)), \tag{2.18}$$

for all $b = 0, \cdots, L_B - 1$, also minimizing the overall cost. The algorithm continues by iterating the three steps until the overall cost converges to a local minimum. The image models constructed using the proposed algorithm for *Spoke* and *Pentagon* in Fig. 2.1 are shown in Fig. 2.5. In these examples, we use $L_C = 9$ determined empirically for the two images as a compromise between the model complexity and accuracy.

2.2.4 Parameter Estimation

To solve (2.16) in Step 1 of the clustering algorithm, we discretize the parameters θ, λ_a, λ_b, ω_s and r_s, similar to our previous work in [28], and estimate the parameter set approximating the solution of (2.16). Because a texture source is composed of an irregular texture component and a regular texture component, as indicated in (2.9), $\sigma_{i|\Theta}^2$ in (2.15) derived from the texture source can also be decomposed into two corresponding components, i.e., $\sigma_{i|\Theta}^2 = \beta\sigma_{i|\Theta^I}^2 + (1-\beta)\sigma_{i|\Theta^R}^2$, where $\sigma_{i|\Theta^I}^2$ and $\sigma_{i|\Theta^R}^2$ result from the irregular and the regular components respectively. Given the choices of θ, λ_a, λ_b, ω_s and r_s, $\sigma_{i|\Theta^I}^2$ and $\sigma_{i|\Theta^R}^2$ for every filter included in (2.15) can be computed, and the optimal β is determined by setting the derivative of the objective in (2.16) with respect to β to zero. Therefore, the optimal β is a function of the other parameters, and finding the optimal parameter set involves searching in a 5-D discrete parameter space.

In the first iteration of the clustering algorithm, instead of performing a full search



(a) *Spoke*, $\alpha(b)$ (b) *Spoke*, $m(b)$ (c) *Spoke*, $\Phi_{tt}(e^{j\Omega})$

(d) *Pentagon*, $\alpha(b)$ (e) *Pentagon*, $m(b)$ (f) *Pentagon*, $\Phi_{tt}(e^{j\Omega})$

Figure 2.5: Image model for *Spoke* and *Pentagon*. In (a) and (d), $\alpha(b)$ indicates the relative white-noise level in each 8×8 block with values between 0 and 1, the brighter the higher. In (b) and (e), the 9 masked images indicate the blocks that are assigned to each of the 9 clusters. In (c) and (f), the discrete-space PSD, $\Phi_{tt}(e^{j\Omega})$, of each of the 9 texture sources representing the 9 clusters is plotted in logarithmic scale.

of the 5 parameters, we conduct the following initialization procedure. First, for each cluster we measure the sample autocorrelation function, denoted by $r_c^*[\mathbf{d}]$, with a small extend of $\mathbf{d}$, from image neighborhoods centered in blocks contained in the cluster. The angle θ is initialized as the direction with the largest average squared gradient in $r_c^*[\mathbf{d}]$, and a sequence of autocorrelation values along θ and another sequence orthogonal to θ are interpolated from $r_c^*[\mathbf{d}]$ and denoted by $\{r_{c,a}^*[d]\}$ and $\{r_{c,b}^*[d]\}$ respectively. Second, a joint search of λ_a, ω_s and r_s are carried out by minimizing the sum of squared differences between the non-parametric sequence $\{r_{c,a}^*[d]\}$ and the

corresponding autocorrelation values derived from the parameters by

$$r_{tt,a}[d|\lambda_a, \omega_s, r_s] = \beta \exp(-\lambda_a \frac{d}{f_s}) + (1-\beta) \sum_{k=1}^{K} \gamma(k, r_s) \cos(k\omega_s \frac{d}{f_s}). \tag{2.19}$$

Note that β is again a function of the selected λ_a, ω_s and r_s. Finally, given the other parameters, λ_b is initialized similarly by minimizing the sum of squared differences between $\{r^*_{c,b}[d]\}$ and the corresponding values computed by

$$r_{tt,b}[d|\lambda_b] = \beta \exp(-\lambda_b \frac{d}{f_s}) + (1-\beta). \tag{2.20}$$

After the initialization procedure, a joint search on $\{\theta, \lambda_a, \lambda_b, \omega_s, r_s\}$ is performed in the vicinity of the initial values to further refine the estimation. In the subsequent iterations, the initial values can either be the parameters estimated in the previous iteration or that obtained by applying again the initialization procedure, whichever leads to a smaller cost. In each iteration, we only find a suboptimal solution of (2.16) due to the discretization and the reduced search space for the refinement. Nonetheless, as long as the total cost in each cluster decreases, so does the overall cost, and the iterative clustering algorithm continues to converge to a local minimum.

2.3 Transform Coding Analysis

Having discussed the construction of the proposed image model, in the following sections we present theoretical analyses based on the model that identify the potential improvement in transform coding of images using transforms adapted to image textures instead of a fixed transform.

2.3.1 Transform Coding Gain

The transform coding gain measures the performance gain of applying a particular transform to encode source samples over a reference scheme that encodes the samples directly using scalar quantization and independent coding of the samples at a fixed

high rate [81,185]. Assume that the source samples are jointly Gaussian, the transform is orthonormal and it decomposes the source samples into S subbands, σ_s^2 denotes the variance of subband s, and η_s denotes the ratio between the number of coefficients in subband s and the total number of coefficients. We further assume that each subband is encoded independently from others, and in each subband the coefficients are encoded with scalar quantization and independent coding of the coefficients at a fixed high rate with optimal rate allocation across subbands. The transform coding gain, denoted by G_T for a transform T, is defined as

$$G_T = \frac{\sum_{s=0}^{S-1} \eta_s \sigma_s^2}{\prod_{s=0}^{S-1} (\sigma_s^2)^{\eta_s}}. \tag{2.21}$$

The numerator is the source variance and is proportional to the MSE distortion in the reconstruction using the reference scheme with a given rate. The denominator is a weighted geometric mean of the subband variances, and is also proportional to the MSE distortion in the reconstruction using the transform with a given rate (the average rate across subbands). The transform coding gain is defined as the ratio between the two distortions when the source is coded at the same rate, i.e., the factor of reduction in distortion resulting from the transform. Note that, from (2.21), a larger transform coding gain is achieved if the transform packs more source energy into fewer subbands.

Now we consider Gaussian texture sources containing only the elliptic random field component described in Sec. 2.1.3, with parameters $\lambda_a = -\ln(0.8) f_s$, $\lambda_b = \lambda_a / k_b$ where $k_b = 2$, 4, 8, 16, and 32, and θ from 0° to 90°, modeling image textures with different degrees of directionality, where a larger k_b leads to stronger directionality, and different orientations. The additional assumption that the texture source is Gaussian will be elaborated in the next section (Sec. 2.3.2). We consider only the elliptic random field rather than the composite texture source described in Sec. 2.11 because of its simple parameterization directly indicating the directionality and orientation in image textures. Some examples of the source model are illustrated in Fig. 2.2. Given the source model, we derive the subband variances and the resulting transform coding gain for the 8×8 DCT and the 8×8 KLT, where an unique KLT is derived

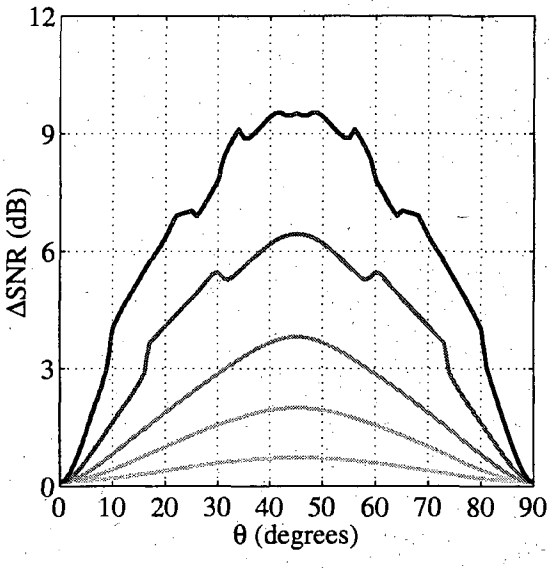


Figure 2.6: Transform coding gain improvement by the 8×8 KLT over the 8×8 DCT. The source is the elliptic random field with $\lambda_a = -\ln(0.8) f_s$, $\lambda_b = \lambda_a / k_b$ where $k_b = 2$, 4, 8, 16, and 32, and θ from 0° to 90°. A darker line corresponds to a larger k_b, i.e., stronger directionality.

for each set of the source parameters. The transform coding gain improvement in decibel (dB) by the KLT over the 2-D DCT, i.e., $10 \cdot \log_{10} \frac{G_{KLT}}{G_{DCT}}$, is plotted in Fig. 2.6. Since the KLT is the optimal transform that maximizes the transform coding gain for a given source and a given transform dimension assuming asymptotically optimal rate allocation [81], this measurement indicates the maximal improvement in reconstruction quality achievable by a transform adapted to the source instead of the fixed 2-D DCT, when the source is coded at the same rate. Note that with the high-rate scalar quantization and independent coding assumption, an improvement of 6.02 dB in the reconstruction quality at the same rate is equivalent to a rate reduction of 1 bit per sample to achieve the same quality [44, 81].

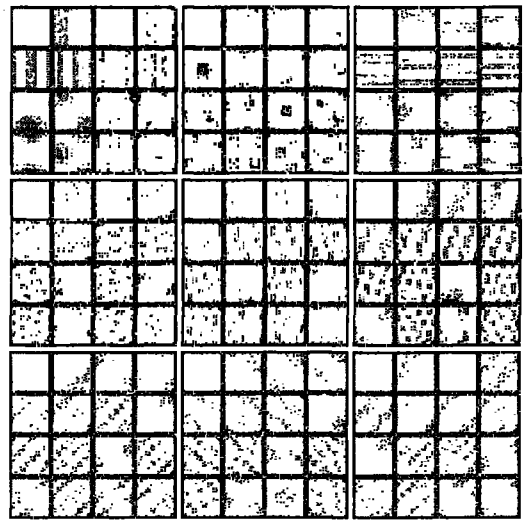
In Fig. 2.6, the improvement increases with the directionality in the source, and the improvement is most significant for directional textures oriented around 45°. For directional textures oriented near 0° and 90°, there is little room for improvement using an adaptive transform and the 2-D DCT is nearly optimal. Notice that the

improvement does not vary smoothly for sources with stronger directionality, as indicated by the notches at the top curves in Fig. 2.6, because of the aliasing components that appear in the source for certain orientations, as discussed in Sec. 2.1.5 and illustrated in Fig. 2.2.

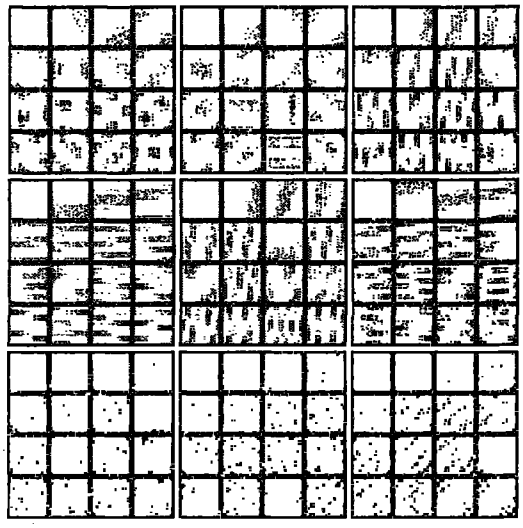
2.3.2 Rate-Distortion Performance

In this section, we analyze the rate-distortion performance for transform coding of images, both for the KLT and the 2-D DCT, with the test images shown in Fig. 2.1. For each image, an image model is first constructed by the texture source clustering algorithm presented in Sec. 2.2. For each constituent texture source modeling the image, the corresponding 8×8 KLT is derived from the parametric autocorrelation function. As an example, the 16 KLT bases having the largest variances for each of the 9 texture sources modeling *Spoke* and *Pentagon*, illustrated in Fig. 2.5, are shown in Fig. 2.7-(a) and Fig. 2.7-(b) respectively. Notice that unlike the 2-D DCT composed of 1-D vertical and horizontal operations, the KLT contains directional bases aligned with the orientation of the texture sources, and these bases are usually among the ones with the largest variances.

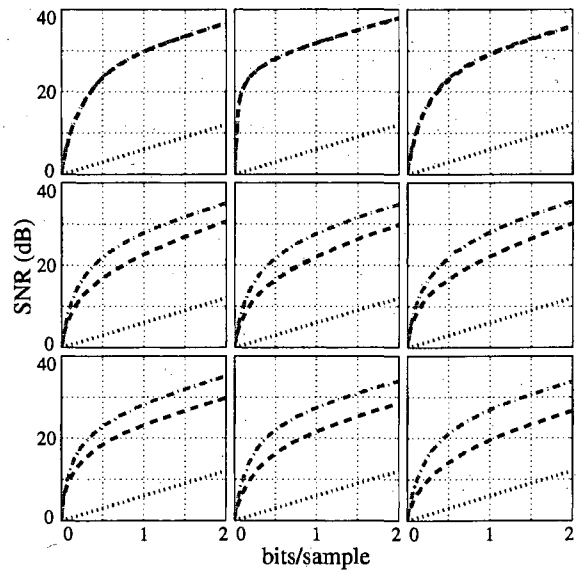
To model the subband coefficients, it has long been observed that for natural images the distribution of a DCT subband (except for the DC subband) across the entire image can be well approximated by a Laplacian distribution [68, 161]. The works in [122] and [118] further showed that within a local region the subband is better approximated by a Gaussian distribution. Additionally, the local variances of the image samples are typically exponentially distributed across the image. As a result, the distribution of a subband over the image can be approximated by a Gauss mixture with an exponential mixing distribution, which is indeed Laplacian. Based on these prior works, we first assume that the subbands in a local image region, e.g., around a block, are Gaussian distributed. Note that the neighborhoods centered in a block is modeled as segments of a realization of the associated texture source, mixed with additive white noise and then scaled in intensity. Therefore, with a further assumption that the additive noise is Gaussian, the underlying texture sources are



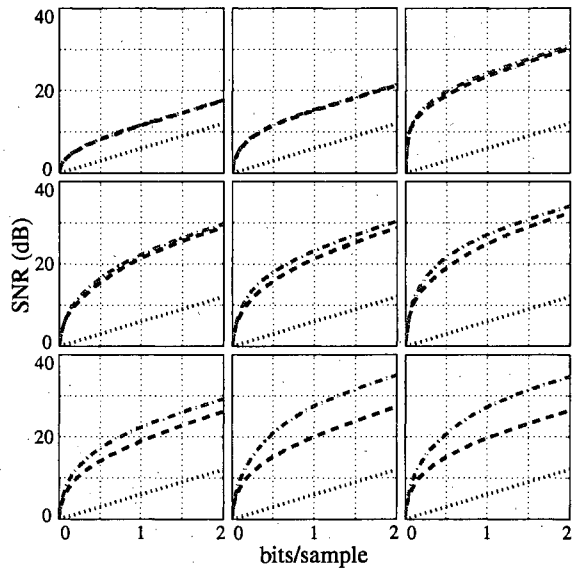
(a) *Spoke*, KLT bases



(b) *Pentagon*, KLT bases



(c) *Spoke*, R-D performance



(d) *Pentagon*, R-D performance

Figure 2.7: Each of the 3×3 plots corresponds to one of the nine texture sources modeling *Spoke* and *Pentagon*, as illustrated in Fig. 2.5. (a) and (b): The 16 bases of the 8×8 KLT having the largest variances, ordered with descending variances from left to right and then top to bottom. (c) and (d): The rate-distortion performance of transform coding for each texture source. In each plot, the top dash-dotted curve represents the performance of the 8×8 KLT, the middle dashed curve represents the 8×8 DCT, and the bottom dotted straight line represents direct coding of the source samples without a transform.

also Gaussian.

We continue the analysis by considering the rate-distortion performance for each unit-variance texture source, now assumed to be Gaussian based on the above arguments, with the 8×8 KLT and the 8×8 DCT. The S subbands, $S = 64$, from the texture source are therefore also Gaussian with variances determined by the source. In this analysis we consider the case that every subband is coded independently from others, and the coding performance achieves the rate-distortion function of Gaussian memoryless sources, i.e.,

$$R_{c,s}(\lambda) = \max\{0, \frac{1}{2}\log_2 \frac{\sigma_{c,s}^2}{\lambda}\}, \quad D_{c,s}(\lambda) = \min\{\lambda, \sigma_{c,s}^2\}, \tag{2.22}$$

where $R_{c,s}$ and $D_{c,s}$ are the rate in bits per coefficient and the MSE distortion for coding subband s in the c-th texture source respectively, $\sigma_{c,s}^2$ denotes the variance of the subband and λ controls the rate-distortion trade-off [44, 185]. Consequently, the total rate and the MSE distortion for coding the c-th texture source using these transforms are computed by

$$R_c(\lambda) = \sum_{s=0}^{S-1} \eta_s R_{c,s}(\lambda), \quad D_c(\lambda) = \sum_{s=0}^{S-1} \eta_s D_{c,s}(\lambda) \tag{2.23}$$

respectively, where η_s, as defined for (2.21), equals $\frac{1}{64}$ for the 8×8 transforms. Note that all subbands share the same λ so that optimal rate allocation among subbands is achieved, and a smaller λ leads to a higher total rate [185]. In Fig. 2.7-(c) for *Spoke* and Fig. 2.7-(d) for *Pentagon*, the performance of the 8×8 KLT and DCT for each texture source is plotted, together with a reference scheme that encodes the source samples directly without a transform, i.e., $S = 1$ and $\sigma_{c,0}^2 = 1$ equals the source variance. In each plot, the rate is represented by bits per source sample, and the distortion is represented by the signal-to-noise ratio (SNR) in decibel (dB), i.e., $-10 \cdot \log_{10} D_c(\lambda)$. Notice that at high rates (small λ) $D_{c,s} = \sigma_{c,s}^2 \cdot 2^{-2R_{c,s}}$, i.e., the SNR increases by 6.02 dB for every increase in rate of 1 bit per sample, and the gap between either the KLT or the DCT performance to the reference scheme is indeed the transform coding gain as defined in (2.21). Furthermore, comparing Fig. 2.7 with

Fig. 2.5, the texture sources with the most improvements by the KLT over the DCT are the ones with stronger directionality and oriented near 45°, consistent with the conclusion in Sec. 2.3.1. These improvements are enabled by the directional bases in the KLT that capture most of the energy in the directional textures.

Following the above analysis for each texture source, we now analyze the image-wise rate-distortion performance. We consider a KLT scheme that operates with L_C modes, each corresponding to the 8×8 KLT derived from one of the L_C texture sources modeling the image. For each block, the KLT scheme adaptively selects the mode corresponding to the texture source associated with the block. The model subband variance around a block, denoted by $\sigma_{b,s}^2$ for subband s and the b-th block, resulting from either the KLT or the DCT, is computed by

$$\sigma_{b,s}^2 = \sigma_b^2(\alpha(b) + (1 - \alpha(b))\sigma_{s|\Theta_{m(b)}}^2), \tag{2.24}$$

where $\sigma_{s|\Theta_{m(b)}}^2$ denotes the subband variance derived from the associated texture source.

Note that the analysis is meant to model the relative performance between the KLT and the DCT in a general framework for transform coding of images, and therefore quantify the potential performance improvement by transforms locally adapted to the image over a fixed transform. It is not intended to model the absolute performance of a particular image coding system. To this end, for coding of the subband coefficients, we still assume independent coding across subbands and in each subband ideal coding achieving the rate-distortion function of Gaussian memoryless sources. The local rate-distortion performance around a block is therefore modeled by

$$R_{b,s}(\lambda) = \max\{0, \frac{1}{2}\log_2 \frac{\sigma_{b,s}^2}{\lambda}\}, \quad D_{b,s}(\lambda) = \min\{\lambda, \sigma_{b,s}^2\}, \tag{2.25}$$

where $R_{b,s}$ and $D_{b,s}$ are the rate and the distortion for coding subband s around the b-th block respectively, and $\sigma_{b,s}^2$ denotes the variance of the subband. Consequently,

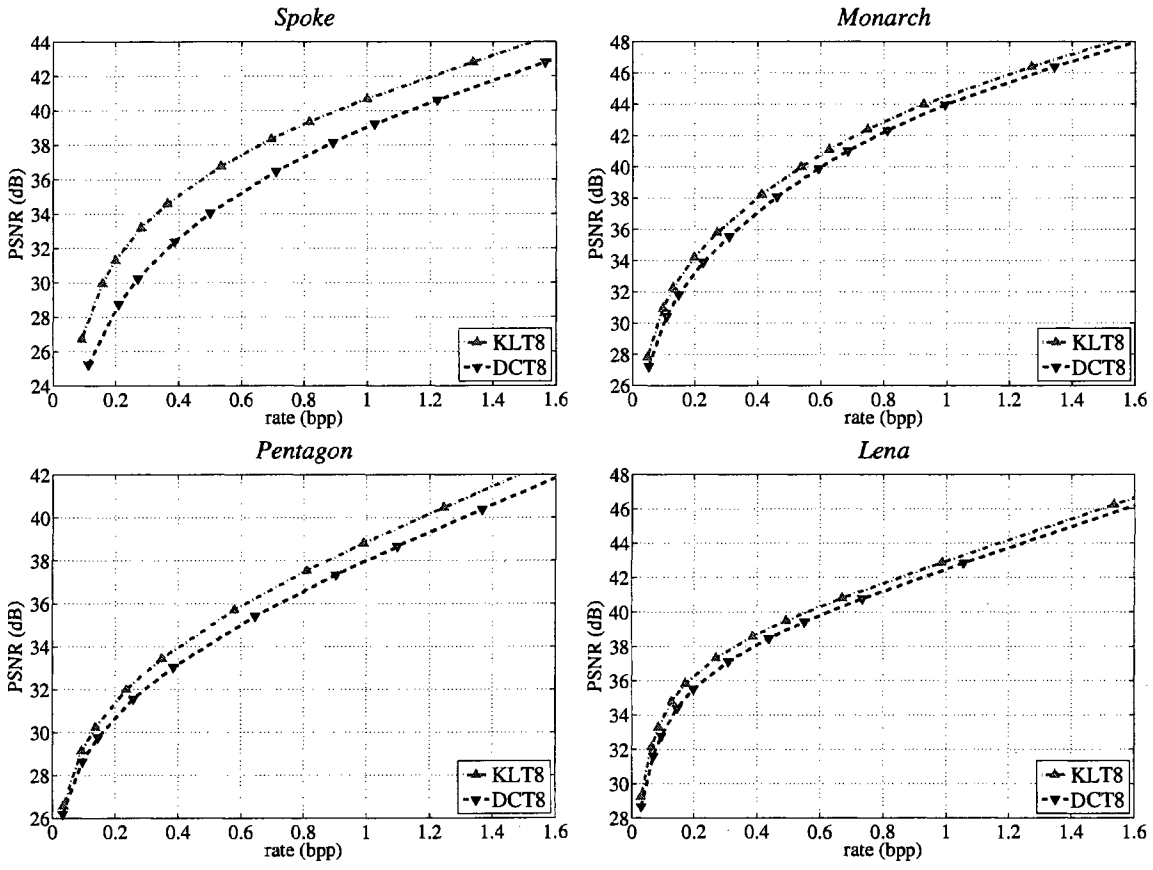


Figure 2.8: Model image-wise rate-distortion performance of the 8×8 KLT and the 8×8 DCT.

the image-wise performance is

$$R(\lambda) = \frac{1}{L_B} \sum_{b=0}^{L_B-1} \sum_{s=0}^{S-1} \eta_s R_{b,s}(\lambda), \quad D(\lambda) = \frac{1}{L_B} \sum_{b=0}^{L_B-1} \sum_{s=0}^{S-1} \eta_s D_{b,s}(\lambda), \tag{2.26}$$

where L_B is the number of blocks in the image. Again, the same λ is used across all subbands and blocks so that optimal rate allocation is achieved. For the test images in Fig. 2.1, the model image-wise rate-distortion performance is plotted in Fig. 2.8, for both the KLT and the DCT. For every image, 9 clusters are used to construct the image model. In each plot, the rate is represented by bits per image sample, or bits per pixel (bpp), and the distortion is represented by the peak signal-to-noise

ratio (PSNR) in decibel, defined as $10 \cdot \log_{10} \frac{255^2}{D(\lambda)}$, a popular measurement for the reconstruction quality of 8-bit grayscale images.

The analysis shows that using a transform adapted to the local statistics, even with only a limited degree of adaptation (9 modes), there is a potential performance gain of up to 3 dB for *Spoke*, 1 dB for *Monarch* and *Pentagon*, and 0.5 dB for *Lena*, over the fixed DCT scheme with the same 8×8 blocks and the same rate. The image-wise performance gain may be less significant for images containing fewer directional features, e.g., *Lena*. Nonetheless, locally around sharp directional features, especially those oriented close to 45°, we can still expect a gain in the order of 5 to 10 dB as suggested by the texture-wise performance in Fig. 2.7-(c) and Fig. 2.7-(d), rendered by the directional bases. Note that in this theoretical analysis, we neglect the rate overhead required to signal the the constituent KLT, or equivalently the texture source parameters, and the mode selections, i.e., the association between the blocks and the texture sources. In practical image coding systems, this overhead for using an adaptive transform should be considered as we shall address in the following chapters.

2.4 Summary

We have proposed to model an image by a set of texture sources to cope with locally varying statistics. Each texture source is a 2-D stationary random field composed of an irregular and a regular texture component, represented by the elliptic random field and the periodic-wave random field respectively. Using the texture sources, any image neighborhood centered in a particular image block is modeled as a 2-D segment of a realization of the texture source associated with the block, mixed with additive white noise and normalized by the local variance and the local mean around the block. For a given image, to jointly determine the parameters of the texture sources modeling the image, the association between the image blocks and the texture sources, and the white-noise level in each block, an iterative clustering algorithm is proposed to optimize the image model for the analysis of transform coding performance.

Using the model, we first consider image textures with different degrees of directionality and orientations, and evaluate the transform coding gain improvement by

the KLT adapted to the textures over the fixed 2-D DCT. The results show that an adaptive transform brings more improvements for textures with stronger directionality, and in particular for directional textures oriented further away from the vertical and the horizontal direction. Rate-distortion analysis based on the image model is also performed for a set of test images. The analysis indicates that using a transform adapted to local statistics in the image instead of a fixed transform, there is a substantial performance gain for images rich of edges and lines. For other images, significant improvements can still be expected locally around sharp directional features, rendered by the directional bases of the adaptive transform aligned with the image features.

Chapter 3

Direction-Adaptive Partitioned Block Transform

In the previous chapter, we have shown that image coding performance can be improved substantially using the KLT locally adapted to the image statistics over a fixed transform. The improvement is mainly contributed from the directional bases of the KLT aligned with directional image features. This suggests that instead of using the KLT, the performance improvement may be rendered by a practical transform with a number of modes, each providing a set of directional bases, that can be adaptively selected according to the local directionality in the image. In this chapter, we propose such a direction-adaptive transform constructed using simple 1-D DCTs, referred to as the direction-adaptive partitioned block transform (DA-PBT). For image coding, the best mode of the DA-PBT is selected at the encoder for each image block using a rate-distortion optimized framework and signaled to the decoder as side information.

In Sec. 3.1, we describe different modes of the DA-PBT, each mode defining a set of transform directions, a block partitioning scheme, coefficient ordering for entropy coding and a quantization matrix. The application of the DA-PBT to image coding in practice is discussed in Sec. 3.2. The proposed transform can also be combined with block-based predictive coding widely adopted in image and video coding techniques, as explained in Sec. 3.3. Theoretical analysis of the coding performance using the DA-PBT based on the image model presented in the previous chapter is included in

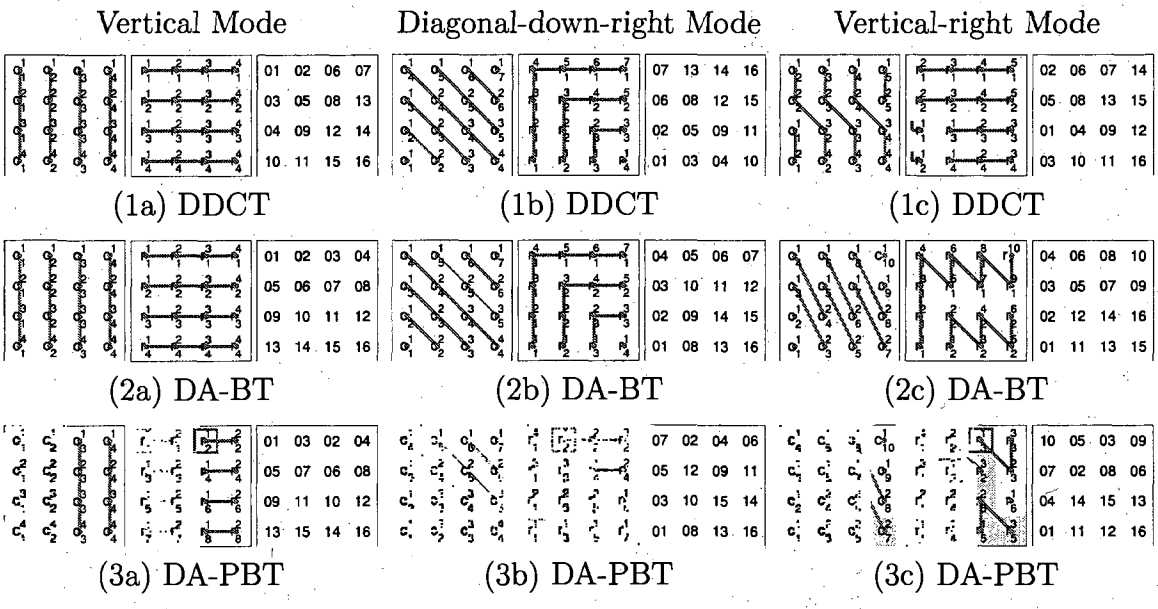


Figure 3.1: The transform directions and the encoding order of the DDCT, the DA-BT and the DA-PBT. In (1c), r_1^1 and r_2^1 are also connected to r_1^2 and r_2^2 respectively, but the lines are omitted for clarity.

Sec. 3.4. Finally, experimental results demonstrating the superior performance of the proposed transform are shown in Sec. 3.5.

3.1 Construction of DA-PBT

To efficiently represent blocks containing directional features, the proposed DA-PBT combines a direction-adaptive block transform (DA-BT) with a direction-adaptive block partitioning scheme, to be discussed in Sec. 3.1.1 and Sec. 3.1.2 respectively.

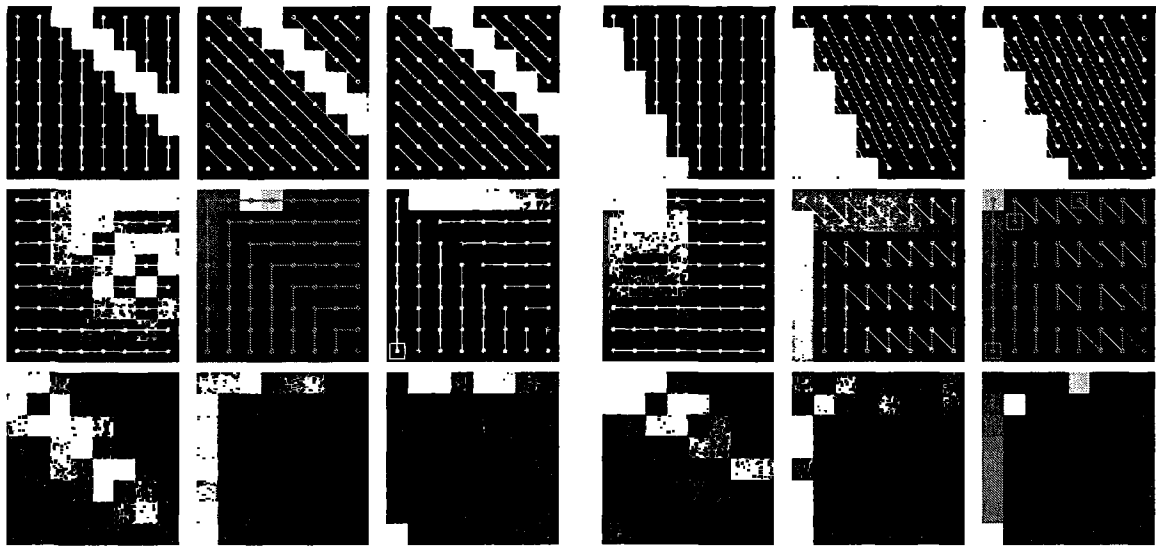
3.1.1 Direction-Adaptive Block Transform

In conventional image coding, the 2-D DCT is composed of two stages of separable 1-D transforms. Taking the 4×4 DCT as an example, the four columns in the 4×4 block shown in Fig. 3.1-(1a) are first transformed in Stage 1 into columns of DCT coefficients, $\{c_i^{1\sim4}\}$, $i = 1, \cdots, 4$. Subsequently, in Stage 2 the rows of coefficients are further transformed into $\{r_j^{1\sim4}\}$, $j = 1, \cdots, 4$. To encode the DCT coefficients, a

zigzag scanning order is adopted, as indicated in the rightmost plot in Fig. 3.1-(1a), so that the coefficients generally decrease in magnitude along the scan. This property can then be exploited in entropy coding of the quantization indices [109, 140], as we shall discuss in Sec. 3.2.2.

To our knowledge, the work most closely related to our proposed DA-PBT is the directional DCT (DDCT) [218,219]. The DDCT consists of directional modes defined similarly as the intra-prediction modes of H.264 [206], in addition to the conventional 2-D DCT. The non-directional mode, the vertical and the horizontal mode in the DDCT all correspond to the conventional transform (Fig. 3.1-(1a)), whereas the other modes, however, are constructed by varying the transform directions. For instance, as shown in Fig. 3.1-(1b), in Stage 1 of the diagonal-down-right mode, the 1-D DCTs, possibly with different lengths, are applied to the sequences along the corresponding direction. In Stage 2, another set of the 1-D DCTs is further applied to the set of coefficients from Stage 1 with the same superscript index, e.g., $\{c^1_{1\sim 7}\}$, because they represent the frequency components at similar spectral locations and are likely to exhibit higher correlation [219]. To encode the resulting coefficients, a zigzag scanning order modified from that in Fig. 3.1-(1a) is adopted, as shown in Fig. 3.1-(1b). Similarly, the transforms and the scanning order in the vertical-right mode of the DDCT are illustrated in Fig. 3.1-(1c). Note that the Stage-1 direction in the DDCT does not match exactly with that defined in the vertical-right mode of H.264, i.e., with a slope $\frac{\Delta y}{\Delta x} = -2$. The remaining modes can be derived by flipping the operations in the diagonal-down-right or the vertical-right mode in the appropriate dimensions.

There are two key differences between the proposed DA-BT and the DDCT. The first difference is in the choice of the transform directions in the vertical-right mode (and the three associated modes with flipped directions). We argue that the direction in Stage 1 of the DDCT (Fig. 3.1-(1c)) is still close to the vertical direction, and therefore the vertical-right mode may not render much benefit over the non-directional or the vertical mode. To provide a directional selectivity that covers all possible feature orientations more evenly, we adopt the configuration in Fig. 3.1-(2c). Aside from the above consideration, the adopted direction matches that in H.264 [206], allowing a



(a) DCT (b) DA-BT (c) DA-PBT (d) DCT (e) DA-BT (f) DA-PBT

Figure 3.2: Transform coefficient magnitude of 8×8 image blocks. The conventional 2-D DCT is used in (a) and (d). For the DA-BT and the DA-PBT, the diagonal-down-right mode is used in (b) and (c) and the vertical-right mode is used in (e) and (f). The original block is shown at the top, together with the Stage-1 directions. The magnitude of the coefficients resulting from Stage 1 is shown in the middle, together with the Stage-2 directions, and the magnitude of the final coefficients is shown at the bottom. In (c) and (f), the coefficients involved in Stage 3 of the DA-PBT are indicated by the squares.

simpler implementation when the DA-BT is combined with intra prediction, as we shall discuss in Sec. 3.3.1.

The second difference is not in the transform itself, but in the ordering when encoding the coefficients. Consider the diagonal-down-right mode of the DDCT shown in Fig. 3.1-(1b). In general, this mode is selected if edges in the corresponding orientation appear in the block. Since the 1-D transforms in Stage 1 are aligned with the edges, energy is concentrated towards the DC coefficients, i.e., $\{c_{1\sim7}^{1}\}$, and the sequence containing the DC coefficients approximately consists of one or multiple step transitions. Consequently, after Stage 2, most energy still resides in $\{r_{1}^{1\sim7}\}$ and the low-frequency components are more likely to have larger magnitude due to the spectral characteristics of the step function. Therefore, we propose the encoding order shown in Fig. 3.1-(2b), which is different from the modified zigzag order used in the

DDCT (Fig. 3.1-(1b)). With the same argument, instead of applying the same operations for the vertical (or horizontal) mode and the non-directional mode as in the DDCT, the encoding order of the vertical mode is modified as shown in Fig. 3.1-(2a), although the transform remains the same.

In addition to the non-directional mode that corresponds to the 2-D DCT, the DA-BT consists of eight directional modes: vertical, vertical-right, diagonal-down-right, horizontal-down, horizontal, horizontal-up, diagonal-down-left, and vertical-left, all can be derived from the configurations in Fig. 3.1. The configurations for 4×4 blocks can be directly extended to 8×8 or 16×16 blocks. To illustrate the performance of the DA-BT, an 8×8 block containing diagonal edges is used as an example and the coefficient magnitude resulting from the conventional 2-D DCT and the DA-BT are shown in Fig. 3.2-(a) and Fig. 3.2-(b) respectively. In this example, many of the 2-D DCT coefficients still retain large magnitude. Additionally, the coefficient magnitude does not generally decrease along the zigzag order, making entropy coding designed based on this assumption inefficient. On the contrary, the DA-BT concentrates the energy to the coefficients located at the first column and the first row while keeping the others zero. It is also evident that the encoding order proposed in Fig. 3.1-(2b) better exploits the distribution of the coefficients than the order adopted in DDCT (Fig. 3.1-(1b)).

Note that in the 2-D DCT, the DC level of a block only affects the DC coefficient, i.e., r_1^1 in Fig. 3.1-(1a). However, for the DA-BT as well as the DDCT, the DC level may contribute to other coefficients, e.g., $\{r_1^{2\sim7}\}$ in Fig. 3.1-(1b), due to the unequal lengths of the Stage-1 transforms as discussed in the shape-adaptive DCT literature [112]. This leakage of the DC energy into the non-DC coefficients hampers energy concentration. To eliminate the problem, the DC separation procedure proposed in [112] is adopted in both the DDCT and the DA-BT. To transform a block, the block mean, denoted by m, is first subtracted from all pixels. After two stages of the transform, the DC coefficient r_1^1 is then set to $\sqrt{N}m$ where N denotes the number of pixels in the block, equal to the DC coefficient of the 2-D DCT. It has been shown in [112] that the resulting transform is reversible using an additional correction procedure. With DC separation/correction, the DC level affects only the

DC coefficient, and, in particular, a constant block leads to at most one nonzero coefficient.

3.1.2 Direction-Adaptive Block Partitioning

Compared to the 2-D DCT, one disadvantage of both the DDCT and the DA-BT is the increase in the maximum length of the constituent 1-D DCTs. For an $S \times S$ block ($S > 2$), the 2-D DCT requires length-S 1-D DCTs, whereas the DDCT and the DA-BT require length-$(2S-1)$ (Fig. 3.1-(1b)) and length-$(3S-2)$ DCTs (Fig. 3.1-(2c)) respectively. To reduce the complexity of the DA-BT, we propose to further divide the block into directional partitions, indicated by different shades in Fig. 3.1-(3b) and Fig. 3.1-(3c). The partition boundaries are aligned with the transform direction in Stage 1, and we limit the Stage-2 transforms so that they do not extend across partition boundaries. In fact, with the proposed partitioning that divides a block into 2 partitions for the diagonal-down-right mode (Fig. 3.1-(3b)) and 3 partitions for the vertical-right mode (Fig. 3.1-(3c)), the maximum required length is S, equal to that of the 2-D DCT. To further exploit the correlation among partitions, a Stage-3 transform is applied to the DC coefficients resulting from Stage 2, indicated by the squares in Fig. 3.1-(3). Additionally, the encoding order is modified to cope with the partitioning. The sequences that originally belong to the same sequence in the un-partitioned DA-BT are encoded in an interleaved manner. For instance, in Fig. 3.1-(3b), $\{r_3^{1\sim3}\}$ and $\{r_4^{1\sim2}\}$ are ordered as r_3^1, r_4^1, r_3^2, r_4^2 and r_3^3. We refer to the partitioned version of the DA-BT as the DA-PBT.

To accommodate the addition of the partitioning and the Stage-3 transform, we propose a two-level DC separation/correction procedure similar to the one-level procedure in [112]. To apply the transform, the block mean, μ, is first subtracted, and the mean in each block-mean-removed partition, denoted by μ_p, $p = 1 \cdots P$ where P is the number of partitions, is further removed, followed by the Stage-1 and Stage-2 transforms. The resulting Stage-2 DC coefficients, e.g., r_1^1 and r_2^1 in Fig. 3.1-(3b), are then set to $\sqrt{N_p}\mu_p$ where N_p is the number of pixels in the corresponding partition. Finally, the Stage-3 transform is applied and the resulting DC coefficient is replaced

by $\sqrt{N}\mu$ where N is the number of pixels in the block.

To reconstruct the block, the DC coefficient is first replaced by 0 followed by the inverse Stage-3 transform. A DC correction procedure similar to that in [112] is then applied to correct the resulting Stage-2 DC coefficients. After performing the inverse Stage-2 transforms in each partition, the Stage-1 DC coefficients, e.g., $\{c^1_{1\sim 7}\}$ in Fig. 3.1-(3b), are then corrected, followed by the inverse Stage-1 transforms. This DC separation/correction procedure ensures the reversibility of the transform. Furthermore, a constant block results in at most one nonzero coefficient after three stages of the transform, and a constant partition also leads to at most one nonzero coefficient in the partition, an additional property achieved through the two-level procedure. Note that the procedure causes a slight deviation from the orthonormality of the transform. Quantitatively, for unit-variance white noise in the transform coefficients, the consequent noise variance in the reconstructed block is below 1.008, 1.003, and 1.001 for 4×4, 8×8, and 16×16 blocks respectively for all the modes in the DA-PBT.

The proposed partitioning not only reduces the complexity, but also improves energy concentration. For instance, consider the image block in Fig. 3.2-(c) where the edges are contained only in the upper-right partition of the diagonal-down-right mode in the DA-PBT. The energy of the DA-PBT coefficients is mostly confined in the upper-right partition whereas the other partition consists of only one nonzero coefficient, exhibiting improved concentration compared to the DA-BT and the 2-D DCT. A similar observation can be made with the image block and the corresponding coefficients in Fig. 3.2-(d)–(f). Moreover, the partitioning also improves visual quality of lossy reconstructions because the compression artifacts generally appear only in the partition containing the edges rather than the whole block. To further improve energy concentration and visual quality for vertical and horizontal image features, the partitioning can be extended to the vertical and the horizontal mode by dividing the block into two rectangular partitions as illustrated in Fig. 3.1-(3a).

The basis functions of the 8×8 DA-PBT for different modes are shown in Fig. 3.3, together with the magnitude of the corresponding frequency responses. In the directional modes, it is clear that the bases are divided into multiple sets, each responsible for a partition in the block. As discussed, this helps to confine the energy of image

features spatially within the associated set of coefficients. Additionally, the directional frequency bands resulting from the directional bases also help to confine the energy of directional features spectrally in a few coefficients.

3.1.3 Quantization

Given the quantization stepsize Q that controls the quality of the reconstruction, a transform coefficient c is mapped to the quantization index η by

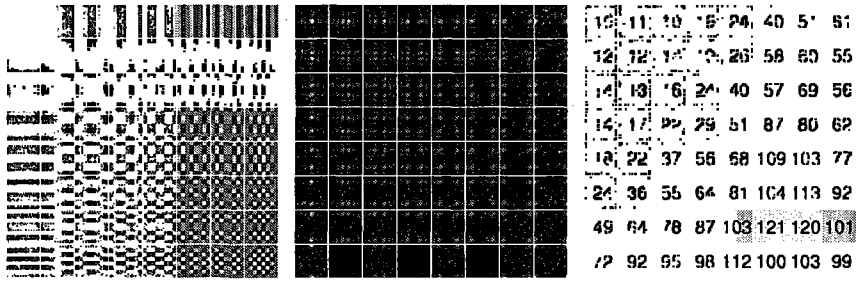
$$\eta = \begin{cases} \text{sign}(c)\lfloor \frac{|c|}{Q} + \Delta \rfloor, & \frac{|c|}{Q} + \Delta > 0, \\ 0, & \text{otherwise}, \end{cases} \tag{3.1}$$

where $\Delta < 1$, and then reconstructed to the corresponding representative level $\hat{c}_\eta$. In JPEG, $\Delta = \frac{1}{2}$ and $\hat{c}_\eta = \eta Q$ so that every quantization interval has the same size and the representative levels are always located in the middle of a quantization interval [109]. In general, a different Δ may be used to adjust the size of the zero interval, i.e., the interval containing zero, and the representative levels can be specified by

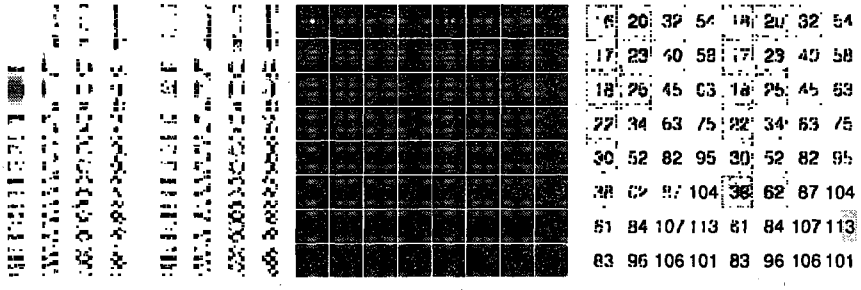
$$\hat{c}_\eta = \begin{cases} \text{sign}(\eta)(|\eta| - \Delta + \xi)Q, & \eta \neq 0, \\ 0, & \eta = 0, \end{cases} \tag{3.2}$$

where $0 \leq \xi < 1$ determines the placement of $\hat{c}_\eta$ within the quantization interval. For example, to better suit the skewed distribution of the transform coefficients, the H.264 reference software uses $\Delta = \frac{1}{3}$ and $\xi = \Delta$ so that the zero interval is larger than the others and the representative levels of the nonzero intervals are closer to the boundary with the lower value [202]. In our implementation, for simplicity we directly adopt this setting without further optimization.

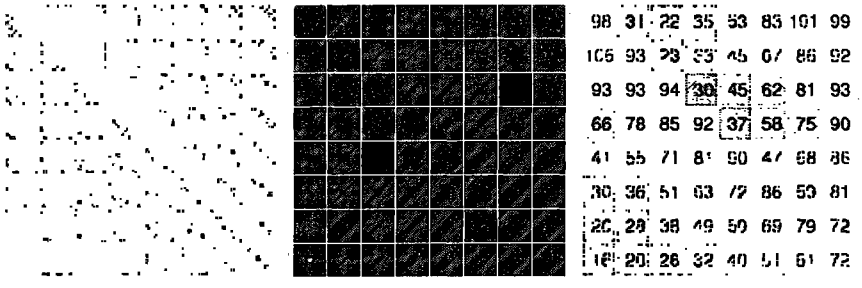
Human visual perception tends to be less sensitive to the amplitude change in the high-frequency patterns, such as the 2-D DCT basis functions corresponding to the high-frequency coefficients shown in Fig. 3.3-(a) [3,131]. Consequently, for a fixed-rate budget, these coefficients can bear more quantization noise than the others. To take advantage of this property, JPEG has suggested 8×8 quantization matrices, obtained



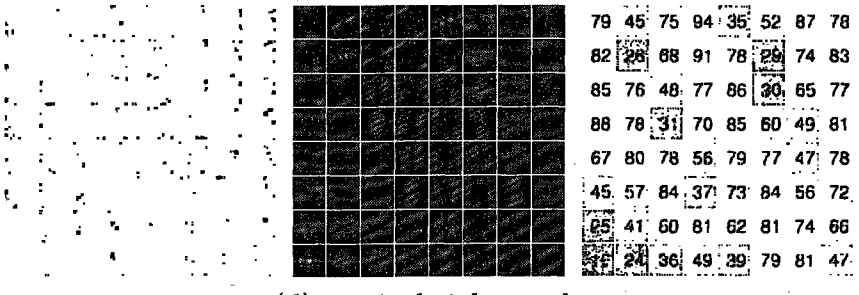
(a) non-directional mode (2-D DCT)



(b) vertical mode



(c) diagonal-down-right mode



(d) vertical-right mode

Figure 3.3: For each mode of the 8×8 DA-PBT, the plots from left to right show the basis functions and the magnitude of the corresponding frequency responses of the forward transform, and the quantization matrix (luminance).

via subjective evaluations as illustrated in the rightmost plot of Fig. 3.3-(a) for the luminance component, that can be scaled to determine the quantization stepsizes for different coefficients [109].

To design the quantization matrices for the eight directional modes of the DA-PBT, we propose the following approach. Assume that a quantization matrix for the 8×8 2-D DCT, where the 64 entries in the matrix are denoted by $\mathbf{q}_c \in \mathbb{R}^{64}$, is already available and achieves the best visual quality in the reconstruction. The quantization matrix suggested in the JPEG standard [109] shown in 3.3-(a) could be used for that purpose. Our goal is to choose the quantization matrix for a directional transform, represented by $\mathbf{q}_d \in \mathbb{R}^{64}$, such that the covariance matrix of the quantization noise in the reconstructed block is close to that resulting from the conventional transform. Let the random variable $\mathbf{x} \in \mathbb{R}^{64}$ represent the pixels in a block, and $\mathbf{T}_c$ and $\mathbf{T}_d \in \mathbb{R}^{64\times 64}$ denote the transform matrix of the conventional and the directional transform respectively such that $\mathbf{T}_c\mathbf{x}$ and $\mathbf{T}_d\mathbf{x}$ represent the corresponding transform coefficients. Additionally, denote the quantization noise in $\mathbf{T}_c\mathbf{x}$ and $\mathbf{T}_d\mathbf{x}$ by $\mathbf{n}_c$ and $\mathbf{n}_d$ respectively.

To simplify the problem, we assume fine quantization such that $E[\mathbf{n}_c\mathbf{n}_c^T]$ and $E[\mathbf{n}_d\mathbf{n}_d^T]$ are diagonal [164, 177], i.e., the transform-domain quantization noise is uncorrelated, and the diagonal entries are proportional to the squared value of the corresponding quantization stepsizes. To meet the aforementioned objective, we minimize the sum of squared differences between the entries in the two covariance matrices of the noise in the reconstruction, i.e., $\mathbf{T}_c^{-1}E[\mathbf{n}_c\mathbf{n}_c^T]\mathbf{T}_c^{-T}$ and $\mathbf{T}_d^{-1}E[\mathbf{n}_d\mathbf{n}_d^T]\mathbf{T}_d^{-T}$, where $\mathbf{T}_c^{-T}$ and $\mathbf{T}_d^{-T}$ denote the transpose of $\mathbf{T}_c^{-1}$ and $\mathbf{T}_d^{-1}$ respectively. Denote $D_2(\mathbf{q}_c)$ and $D_2(\mathbf{q}_d)$ as the diagonal matrices where each diagonal entry is the squared value of the corresponding element in $\mathbf{q}_c$ and $\mathbf{q}_d$ respectively, and

$$\mathbf{C}_c = \mathbf{T}_c^{-1}D_2(\mathbf{q}_c)\mathbf{T}_c^{-T},\ \mathbf{C}_d = \mathbf{T}_d^{-1}D_2(\mathbf{q}_d)\mathbf{T}_d^{-T}. \tag{3.3}$$

Denote the trace of a matrix $\mathbf{T}$ by $tr(\mathbf{T})$, the problem can then be formulated as

$$\underset{\mathbf{q}_d}{\operatorname{argmin}}\ tr((\mathbf{C}_d - \mathbf{C}_c)(\mathbf{C}_d - \mathbf{C}_c)^T). \tag{3.4}$$

Since $\mathbf{T}_d$ is very close to orthonormal as discussed in Sec. 3.1.2, $\mathbf{T}_d\mathbf{T}_d^T \approx \mathbf{I}$ and therefore

$$\begin{aligned} & tr((\mathbf{C}_d - \mathbf{C}_c)(\mathbf{C}_d - \mathbf{C}_c)^T) \\ \approx\ & tr(\mathbf{T}_d\mathbf{T}_d^T(\mathbf{C}_d - \mathbf{C}_c)(\mathbf{C}_d - \mathbf{C}_c)^T\mathbf{T}_d\mathbf{T}_d^T) \\ =\ & tr(\mathbf{T}_d(\mathbf{C}_d - \mathbf{C}_c)\mathbf{T}_d^T\mathbf{T}_d(\mathbf{C}_d - \mathbf{C}_c)^T\mathbf{T}_d^T) \\ =\ & tr((D_2(\mathbf{q}_d) - \mathbf{T}_d\mathbf{C}_c\mathbf{T}_d^T)(D_2(\mathbf{q}_d) - \mathbf{T}_d\mathbf{C}_c\mathbf{T}_d^T)^T). \end{aligned} \tag{3.5}$$

From (3.5), the solution of the the original objective (3.4) can be closely approximated by the square root of the diagonal entries in $\mathbf{T}_d\mathbf{C}_c\mathbf{T}_d^T$. The rounded luminance quantization matrices for the directional modes in the 8×8 DA-PBT derived from the JPEG luminance quantization matrix using this method are shown in Fig. 3.3-(b) to Fig. 3.3-(d). The chrominance matrices can be similarly derived from the matrix suggested in JPEG [109]. For the 4×4 and 16×16 transforms, the 8×8 matrix in JPEG is first downsampled or upsampled to generate the quantization matrices for the conventional transform, and those for the directional transforms are then derived using the same method.

3.2 Image Coding with DA-PBT

3.2.1 Direction Selection

To encode an image, we divide the image into 16×16 macroblocks. Each macroblock may contain a single 16×16 block or four 8×8 blocks, and each 8×8 block can be further divided into 4×4 blocks. Every block is assigned with one of the nine modes of the direction-adaptive transform with the same size as the block. When blocksize 8×8 or 4×4 is selected, a 2×2 or 4×4 2-D DCT is applied to the DC coefficients in the constituent blocks so that only one overall DC coefficient remains in a macroblock. Additionally, to exploit the correlation across macroblocks, an extra 2-D DCT is applied to the DC coefficients in every group of 4×4 macroblocks.

An integer quantization parameter $Q_{\text{H.264}}$ from 0 to 51 that can be directly mapped

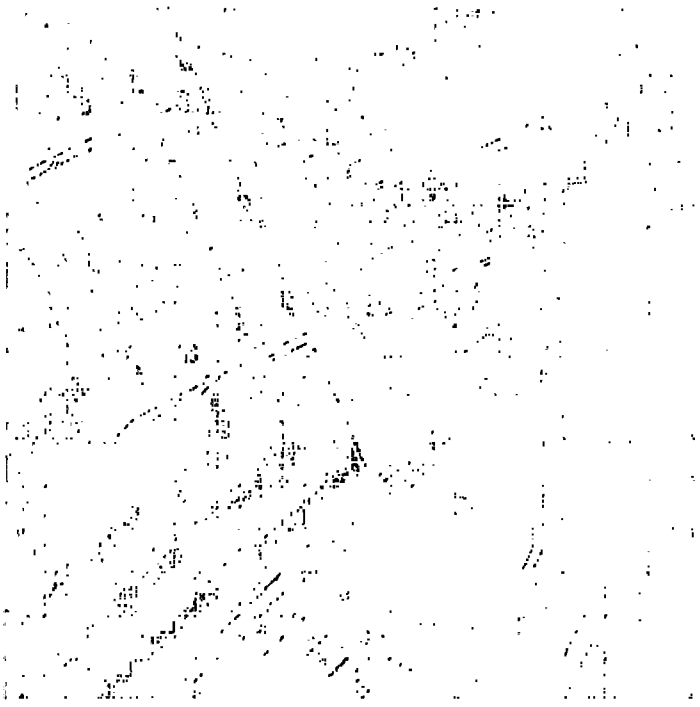


Figure 3.4: The selected blocksizes and modes for a 256×256 region of the *Pentagon* image. The blocksizes are indicated by the square blocks, and the directional modes are indicated by the additional lines delineating the direction-adaptive block partitions. A square block without further partitions represents the non-directional mode. The overhead signaling these selections is coded at 0.04 bpp, around 5% of the total rate.

to a quantization stepsize Q as defined in H.264 is first determined to set the desired reconstruction quality [206]. Given $Q_{\text{H.264}}$, for each macroblock the blocksize and the modes are selected by minimizing a Lagrangian cost function $\bar{D}_c + \bar{\lambda}(\bar{R}_c + \bar{R}_s)$, similar to entropy-constrained vector quantization [35] and the rate-distortion optimized framework for motion estimation in video coding [82, 205]. In the cost function, $\bar{D}_c$ denotes the distortion (sum of squared error) in the reconstructed macroblock, $\bar{R}_c$ and $\bar{R}_s$ denote the number of bits required to encode the quantization indices and the overhead signaling the selection respectively, and $\bar{\lambda}$ is the Lagrangian multiplier set to $0.85 \cdot 2^{(Q_{\text{H.264}}-12)/3}$ obtained empirically in the context of hybrid video coding [204,205].

An example of the selected blocksizes and modes is shown in Fig. 3.4. In our experiments, the overhead signaling the blocksizes and the transform modes typically

takes about 5% of the total rate for most images. The figure also shows that the DA-PBT can be viewed as a generalization of the variable block-size transforms in [207] with the inclusion of non-rectangular partitions and directional transforms. As a post-processing step, the blocking artifacts typically observed in block-transform-based image coding are mitigated by an adaptive deblocking filter modified from that proposed for the 4×4 blocks in H.264 to accommodate the 8×8 and 16×16 blocks [128].

3.2.2 Entropy Coding

To encode the quantization indices, context-based adaptive binary arithmetic coding (CABAC) used in H.264 for 4×4 blocks is adopted and further extended to handle 8×8 and 16×16 blocks [140]. Together with the encoding order illustrated in Fig. 3.1, CABAC exploits the common patterns of trailing 1's and 0's along the ordered indices to improve compression.

The side information signaling the blocksizes and the modes is also encoded using CABAC, similar to encoding the motion information in H.264 [140]. Specifically, one symbol for each macroblock indicates if the macroblock is divided into four 8×8 blocks, and another symbol for each 8×8 block indicates if it is further divided. To encode the mode of a block, one symbol first signals if the selected mode is the non-directional mode. If this is not the case, we first represent each directional mode by a number from 1 for the vertical mode counterclockwise to 8 for the vertical-left mode, and also represent the non-directional mode by 0. Additionally, denote the modes previously selected for the left, the top, the top-left, and the top-right neighbor of the current block, i.e., the blocks in the causal neighborhood, by m_l, m_t, m_{tl} and m_{tr} respectively. If any of these neighbors exists and uses a directional mode, the mode of the current block is predicted from the neighbors, in modulo-8 arithmetic, and only the prediction residual is signaled. Otherwise the mode is signaled directly without prediction. To come up with the prediction, denoted by m_p, we use the following rules. If the left neighbor exists and it has a mode corresponding to a horizontal-ish direction, i.e., $4 \leq m_l \leq 6$, we select $m_p = m_l$ because it is likely that a horizontal-ish

image feature extends from the left neighbor to the current block. If the condition for m_l does not hold, similar conditions are evaluated for m_t, m_{tl} and m_{tr} in order. If none of the above conditions holds, we continue with the following rules. If none of m_l, m_t and m_{tl} is 0, we select $m_p = m_l$ if m_{tl} is closer to m_t than to m_l, again in modulo-8 arithmetic, i.e., the modes are closer between horizontal neighbors than vertical neighbors, and $m_p = m_t$ otherwise. Instead, if any of m_l, m_t and m_{tl} is 0, $m_p = m_l$ if $m_l \neq 0$, $m_p = m_t$ if $m_l = 0$ and $m_t \neq 0$, and otherwise the current mode is directly signaled without prediction.

3.2.3 Computational Complexity

The rate-distortion optimized direction selection process generally requires performing the transform, quantization, and entropy coding for every mode to evaluate the rate and the distortion. Therefore, the computational complexity of image coding with DA-PBT at the encoder is approximately nine times that of conventional image coding using the 2-D DCT. Note that the decoder complexity is about the same as a conventional decoder since only the mode selected at the encoder is performed at the decoder.

To reduce the encoder complexity, one approach is to replace the rate and distortion evaluation with a simpler measurement such as the sum of absolute values (SAV) of the transform coefficients. The original Lagrangian cost $\bar{D}_c + \bar{\lambda}(\bar{R}_c + \bar{R}_s)$ becomes $\bar{S}_c + \sqrt{\bar{\lambda}}\bar{R}_s$ where $\bar{S}_c$ denotes the SAV, a simplification often used in video coding implementation [204,205]. This simplification bypasses the need for quantization and entropy coding of the coefficients for every mode, however, in our experiments, it can lead to significant loss in performance.

Another approach is to examine only a subset of the modes based on the output of a simple classifier, analogous to the technique in classified vector quantization [157] where one out of multiple codes is selected for an image block based on classification of the content instead of testing through all codes as in universal vector quantization [67]. In particular, we include in the selection process only the non-directional mode and at most one directional mode obtained as follows. For each directional

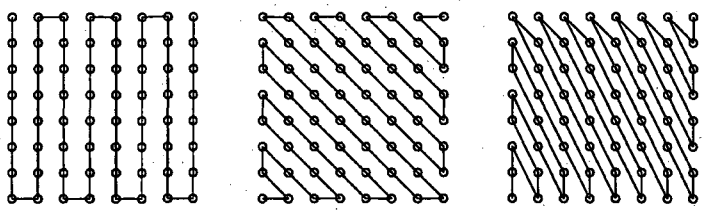


Figure 3.5: Directional zigzag scanning order to estimate the gradient along the (a) vertical, (b) diagonal-down-right, and (c) vertical-right direction.

mode, a directional zigzag scanning order is defined as illustrated in Fig. 3.5 using 8×8 blocks as an example, and the directional gradient is measured as the sum of absolute differences between every two consecutive image pixels along the order, calculated using simple integer operations. The direction with the smallest gradient is referred to as the estimated direction, and the corresponding mode is included in the selection process if the following two conditions hold. First, the largest gradient exceeds a threshold, e.g., four times the number of pixels in the block, so that the block indeed contains significant variations that may be better handled by a directional transform. Second, at least one of the directions with the second or the third smallest gradient is an immediate neighbor to the estimated direction. This condition suggests that the block contains image features with a single orientation, e.g., lines and edges, close to the estimated direction, rather than complex patterns having multiple dominant orientations, e.g., corners, that cannot be exploited efficiently by the directional transform. By considering at most one additional mode, this approach significantly reduces the encoder complexity to less than twice of the complexity of a conventional encoder while incurring only a small performance loss, as shall be presented in Sec. 3.5.

3.3 Residual Coding with DA-PBT

Hybrid coding is a technique widely used in image and video coding [206]. It consists of two steps. In the first step, the block to be encoded is predicted by a prediction

block generated from the reconstruction of the previously encoded pixels. The reconstruction requires a decoder loop to be included at the encoder. Hence, this step is referred to as closed-loop prediction. Depending on the source of these pixels, there are two types of closed-loop prediction: intra prediction, using pixels in the same image, and inter prediction, using pixels in other images, e.g., previously encoded video frames. In the second step of hybrid coding, the residual block is further decorrelated by transform coding, where block transforms are especially suitable because of the block-based prediction. Assuming directional features exist in the block to be encoded, for both intra and inter prediction, directionality may still remain in the residual block due to limited prediction accuracy. This directionality can be exploited by the DA-PBT to improve the overall performance of hybrid coding.

3.3.1 Intra Prediction

Directional intra prediction is included in H.264. To encode the current block, a prediction block is extrapolated along a certain direction from previously encoded pixels. Eight directional modes are defined in H.264 for 4×4 blocks, aligned exactly to the Stage-1 directions in the DA-PBT, together with a non-directional mode (DC mode) where the prediction is simply the average of the surrounding pixels [206]. We argue that the directionality in the original block should be close to the direction selected for intra prediction, and therefore so is the remaining directionality in the residual. Based on this argument, instead of applying a conventional transform to the residual, we propose to always apply the DA-PBT using the same mode as intra prediction. This method requires no additional search for the transform mode, and thus no extra signaling is needed and the computational complexity is approximately the same as if the 2-D DCT is always applied. Moreover, DC separation/correction can be omitted since the DC energy is typically small in the residual. To further reduce the complexity, directional intra prediction and the DA-PBT along the same direction can be combined by applying the DA-PBT first and then performing prediction only to the Stage-1 DC coefficients (with proper scaling), rather than to all pixels in the block. Note that we extend the nine intra prediction modes for 4×4 blocks defined

in H.264 to 8×8 and 16×16 blocks to be applied in conjunction with the DA-PBT of the corresponding blocksize.

3.3.2 Inter Prediction

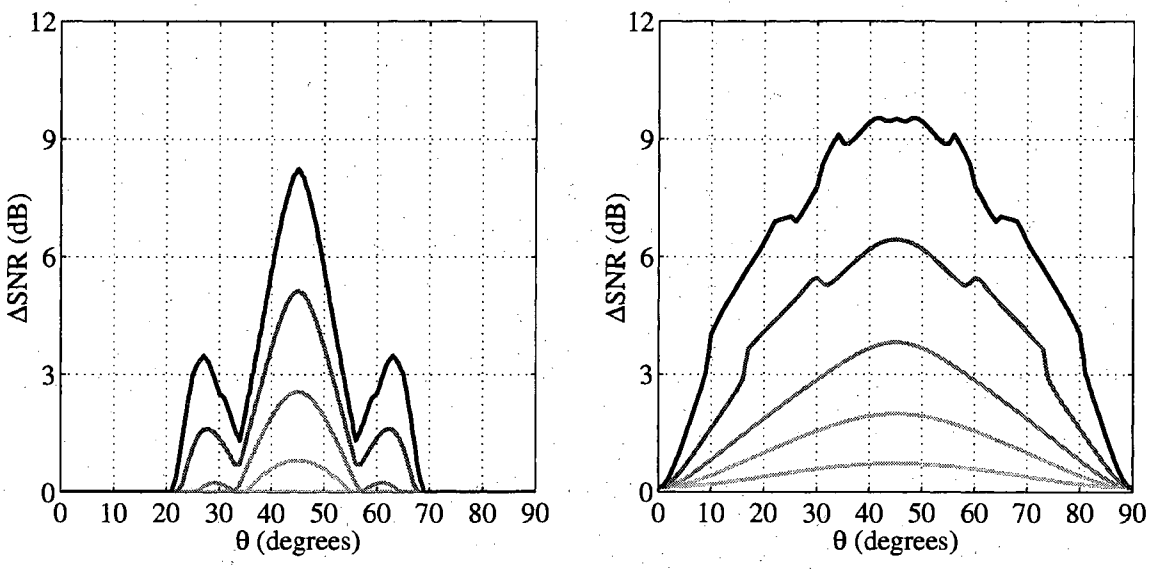
In most video coding standards, inter prediction is achieved through block-based motion-compensated prediction [206]. Analogous to the argument for intra prediction, if directional features exist in the block to be encoded, similar directionality generally resides in the corresponding prediction block, generated from the previously encoded video frames, and possibly also in the residual block. To apply the DA-PBT to the inter residual, instead of considering all the modes, the simplifying approach described in Sec. 3.2.3 is again adopted. However, instead of measuring the directional gradients in the residual block that is to be transformed, the measurement is performed on the prediction block based on the assumption that the two blocks possess similar directionality if directional features still reside in the residual block. Only if the measured gradients satisfy the conditions in Sec. 3.2.3, suggesting high directionality in the prediction block and therefore possibly in the residual, one directional mode of the DA-PBT is considered in the direction selection process at the encoder, in addition to the non-directional mode. At the decoder, these conditions can be examined by performing the same measurement on the prediction block generated during the decoding process. Therefore, only if the conditions are satisfied the decoder needs to decide whether the non-directional mode or the directional mode has been selected, which can be signaled with a binary symbol from the encoder. More importantly, without further overhead, the decoder can recover this directional mode as it is the one with the smallest measured gradient. When applying the DA-PBT to inter residual blocks that already require less rate to encode, this approach greatly reduces the rate overhead that can easily nullify the potential gain of the DA-PBT.

3.4 Theoretical Analysis of DA-PBT

3.4.1 Transform Coding Gain

Similar to the analysis in Sec. 2.3.1, the transform coding gain improvement in decibel by the DA-PBT over the conventional 2-D DCT using 8×8 blocks, i.e., $10 \cdot \log_{10} \frac{G_{DA-PBT}}{G_{DCT}}$, is plotted in Fig. 3.6-(a), and the improvement by the KLT originally shown in Fig. 2.6 is included again in Fig. 3.6-(b) for reference. The source is assumed to be the elliptic random field described with an additional Gaussianity assumption as described in Sec. 2.3.1, modeling image textures with different degrees of directionality and different orientations. For each set of the source parameters, the DA-PBT mode with the highest transform coding gain is adaptively selected. For the DA-PBT the slight deviation from orthonormality discussed in Sec. 3.1.2 is neglected such that the definition of the transform coding gain in (2.21) for orthonormal transforms is still applicable. In Fig. 3.6, we consider only the elliptic random fields oriented from 0° to 90° due to the symmetry between the two sets of the directional modes in the DA-PBT. For instance, an elliptic random field with $\theta = 45°$ is aligned with the diagonal-down-left mode and $\theta = 135°$ is aligned with the diagonal-down-right mode, and the two cases result in the same transform coding gain improvement.

Being the transform that maximizes the coding gain, the improvement by the KLT is certainly always larger than that by the DA-PBT. Nevertheless, using only a limited number of modes and combinations of simple 1-D transforms, the DA-PBT is already able to accomplish certain improvements. In Fig. 3.6-(a), the three peaks from left to right correspond to the vertical-left, diagonal-down-left, and horizontal-up mode respectively, and the improvement from direction adaptation is larger for sources with stronger directionality. Notice that there is no improvement by the DA-PBT for sources oriented near 0° and 90°, i.e., the vertical and the horizontal mode are never selected over the non-directional mode (2-D DCT). From Fig. 3.1-(3a), the vertical mode, for example, essentially contains two 4×8 2-D DCTs. From Fig. 3.3-(a) and Fig. 3.3-(b), the 64 frequency bands in the 8×8 2-D DCT approximately divide each of the 32 bands in the 4×8 transform into two bands, further reducing the weighted geometric mean in (2.21). Therefore, the transform coding gain of the 8×8 2-D DCT



(a) DA-PBT over DCT (b) KLT over DCT

Figure 3.6: Transform coding gain improvement over the 8×8 DCT by (a) the 8×8 DA-PBT, and (b) the 8×8 KLT. The source is the elliptic random field with $\lambda_a = -\ln(0.8) f_s$, $\lambda_b = \lambda_a / k_b$ where $k_b = 2$, 4, 8, 16, and 32, and θ from 0° to 90°. A darker line corresponds to a larger k_b, i.e., stronger directionality.

is never smaller than that of the 4×8 transform and the vertical mode (and the horizontal mode) is never selected in this analysis.

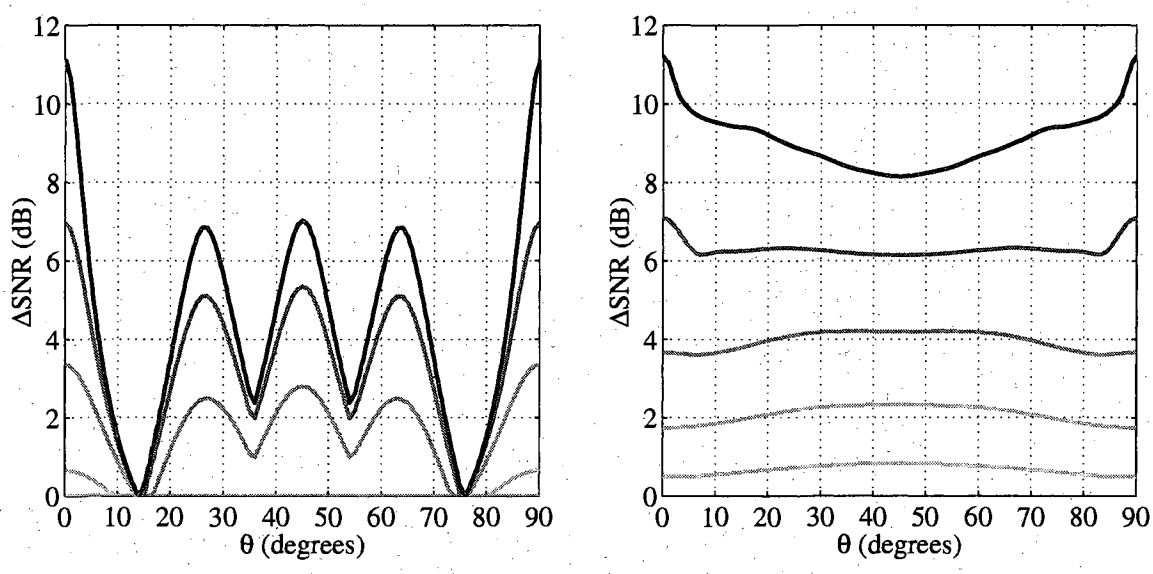
In practice, the performance gain by the DA-PBT comes from firstly the directional frequency bands in the transform that increases energy concentration spectrally, secondly the partitioning scheme that spatially concentrates energy, and thirdly the more efficient encoding order. Only the first factor can be accounted for by the transform coding gain analysis. The second is based on the premise that directional features only appear in some partitions, and hence it cannot be evaluated using the stationary source model assuming same statistics in the entire block. The third factor cannot be captured by the analysis because of the independent coding assumption used in the definition of the transform coding gain discussed in Sec. 2.3.1. To overcome some of the limitations, we further develop the analysis using linear approximation as to be discussed Sec. 3.4.2.

3.4.2 Linear Approximation

Linear approximation measures the MSE distortion in the reconstruction resulting from keeping only a fixed set of the transform coefficients and setting others to zero [134, 196]. In the context of comparing the coding performance of the DA-PBT and the 2-D DCT, we consider the fixed set to be the first M coefficients along the encoder order defined for each mode of the DA-PBT, and along the zigzag order defined for the 2-D DCT. Note that this is different from the nonlinear approximation metric, also referred to as the energy packing efficiency in the analysis in [219], that keeps the M largest coefficients [134, 196].

Linear approximation reflects the combined efficiency of both the transform itself and the encoding order. This is because an efficient transform concentrates the energy into a few coefficients, and these coefficients should be located in the beginning of the encoder order so that the magnitude of the coefficients generally decreases along the order, a desirable property that can be later exploited by the entropy coder as explained in Sec. 3.2.2.

We denote the distortion for the 2-D DCT by D^M_{DCT}. Assume that the DA-PBT always selects the mode with the best linear approximation, and denote the resulting distortion by D^M_{DA-PBT}. Note that the modes are selected adaptively, and in this regard the approximation is no longer linear [134, 196]. Nonetheless, in each mode the preserved coefficients are always the same, and therefore the approximation in this scope is still linear. Using $M = 10$, the linear approximation improvement by the DA-PBT over the 2-D DCT in decibel, i.e., $10 \cdot \log_{10} \frac{D^M_{DCT}}{D^M_{DA-PBT}}$ is plotted in Fig. 3.7-(a). Additionally, the improvement by the KLT is plotted in Fig. 3.7-(b), where the M retained coefficients are naturally the ones having the largest variances, derived from the source model. Notice that, different from the transform coding gain analysis in Fig. 3.6-(a), in Fig. 3.7-(a) the improvements near 0° and 90° from the encoding orders of the vertical mode and the horizontal mode are clearly captured by the linear approximation analysis, and they follow closely to the case using the optimal transform and ordering, i.e., the KLT. The other peaks also indicate that significant improvements are enabled by the vertical-left, the diagonal-down-left, and the horizontal-up mode of the DA-PBT.



(a) DA-PBT over DCT (b) KLT over DCT

Figure 3.7: Linear approximation improvement over the 8×8 DCT by (a) the 8×8 DA-PBT, and (b) the 8×8 KLT. The source is the elliptic random field with $\lambda_a = -\ln(0.8) f_s$, $\lambda_b = \lambda_a / k_b$ where $k_b = 2$, 4, 8, 16, and 32, and θ from 0° to 90°. A darker line corresponds to a larger k_b, i.e., stronger directionality.

3.4.3 Rate-Distortion Performance

Extending from the analysis in Sec. 2.3.2 that models the image-wise rate-distortion performance of the 8×8 KLT and the 8×8 DCT using the proposed image model, in this section we further derive the model performance of the 8×8 DA-PBT. Note that this analysis is intended to model the relative performance of adopting the DA-PBT, instead of the KLT or the 2-D DCT, in a general framework for transform coding of images. It is not meant to model the absolute performance of the proposed image coding system presented in Sec. 3.2.

As discussed in Sec. 2.3.2, for every mode of the DA-PBT, the model subband variances around a block can be derived from the image model. Locally around a block, we assume that every subband is coded independently from others, and the coding performance achieves the rate-distortion function of Gaussian memoryless sources. As a result, numbering the 9 modes of the DA-PBT from $m = 0$ to $m = 8$, the local rate-distortion performance around the b-th block in the image using mode

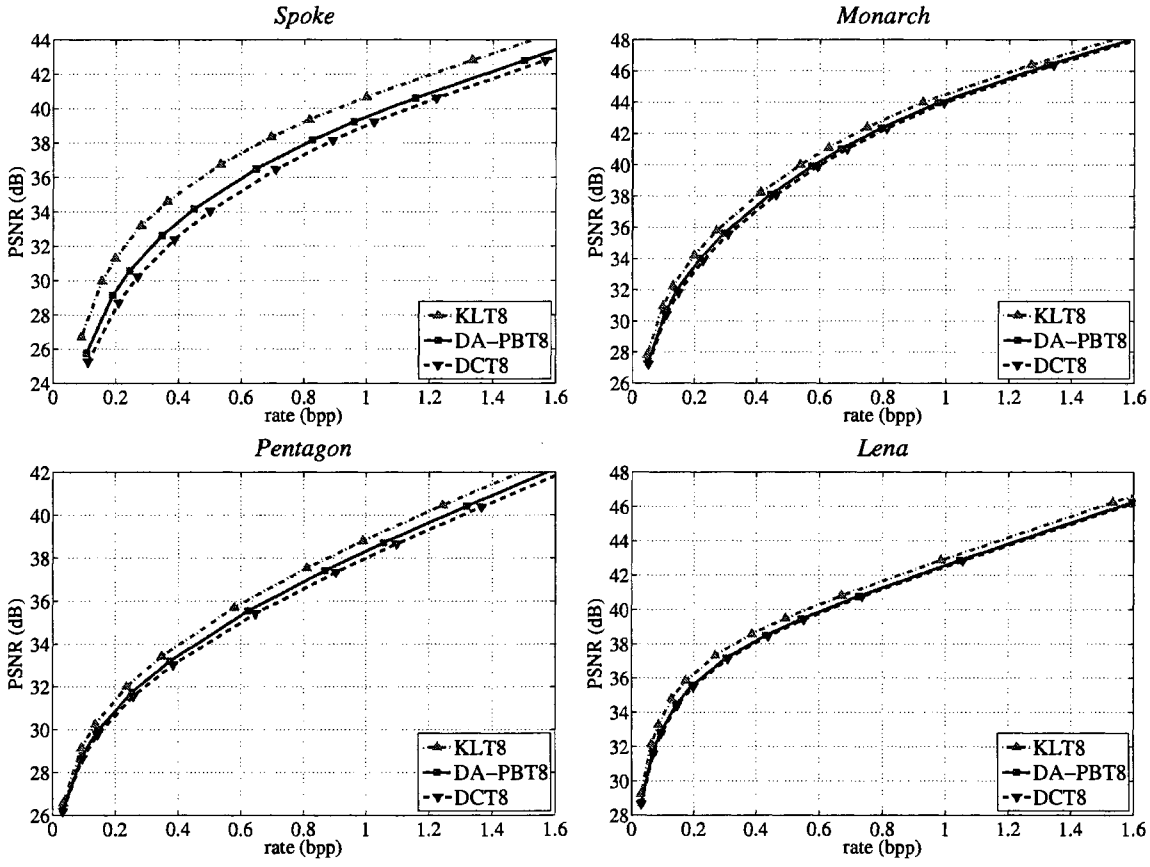


Figure 3.8: Model rate-distortion performance of the 8×8 KLT, the 8×8 DA-PBT and the conventional 8×8 DCT.

m can be expressed as

$$R_b^{(m)}(\lambda) = \frac{1}{64}\sum_{s=0}^{63} \max\{0, \frac{1}{2}\log_2 \frac{\sigma_{b,s}^{2,(m)}}{\lambda}\}, \quad D_b^{(m)}(\lambda) = \frac{1}{64}\sum_{s=0}^{63} \min\{\lambda, \sigma_{b,s}^{2,(m)}\}, \tag{3.6}$$

where $\sigma_{b,s}^{2,(m)}$ denotes the variance of subband s around the b-th block using mode m, and λ controls the rate-distortion trade-off. For this block, the adopted DA-PBT mode is selected by minimizing the Lagrangian cost $D_b^{(m)}(\lambda) + \bar{\lambda} R_b^{(m)}(\lambda)$, same as the formulation in Sec. 3.2.1 but ignoring the signaling overhead in this theoretical analysis, and $\bar{\lambda} = 2\ln 2 \cdot \lambda$ as derived in [185]. Denote the mode selected for the b-th block by m_b and the number of blocks in the image by L_B, the image-wise performance

is then modeled by

$$R(\lambda) = \frac{1}{L_B} \sum_{b=0}^{L_B-1} R_b^{(m_b)}(\lambda), \quad D(\lambda) = \frac{1}{L_B} \sum_{b=0}^{L_B-1} D_b^{(m_b)}(\lambda). \tag{3.7}$$

The model performance of the DA-PBT for the test images shown in Fig. 2.1 is plotted in Fig. 3.8, together with the model performance of the KLT and the 2-D DCT originally shown in Fig. 2.8. Similar to the transform coding gain analysis in Sec. 3.4.1, the model performance of the DA-PBT only captures the improvement from the transform itself. The potential gain from the block partitioning scheme and the improved encoding order is unaccounted for. Not surprisingly, in Fig. 3.8 the performance gain from the DA-PBT is limited compared to the maximum gain achievable by the KLT. Nonetheless, in Sec. 3.5 we shall see that for image coding in practice, the DA-PBT can indeed deliver significant improvements over the 2-D DCT in rate-distortion measurements as well as in perceptive quality.

3.5 Experimental Results

The performance of the DA-PBT is reported in this section. In the experimental results, the adaptive deblocking filter (Sec. 3.2.1) and the simplified search (Sec. 3.2.3) are enabled whereas the quantization matrices (Sec. 3.1.3) are disabled unless specifically mentioned.

3.5.1 Still Image Coding

We first present results using the 512×512 grayscale test images shown in Fig. 2.1. The rate-distortion performance of using various block transforms in image coding with only 8×8 blocks is plotted in Fig. 3.9 including (1) **JPEG**: baseline JPEG with entropy coding defined in [109], without quantization matrices and with the additional deblocking filter in Sec. 3.2.1, (2) **DCT8**: the 2-D DCT, (3) **DDCT8** in [219], (4) **DA-BT8**: the direction-adaptive transform described in Sec. 3.1.1 without partitioning, (5) **DA-PBT8**, and (6) **DA-PBT8-full**: DA-PBT not using the simplified

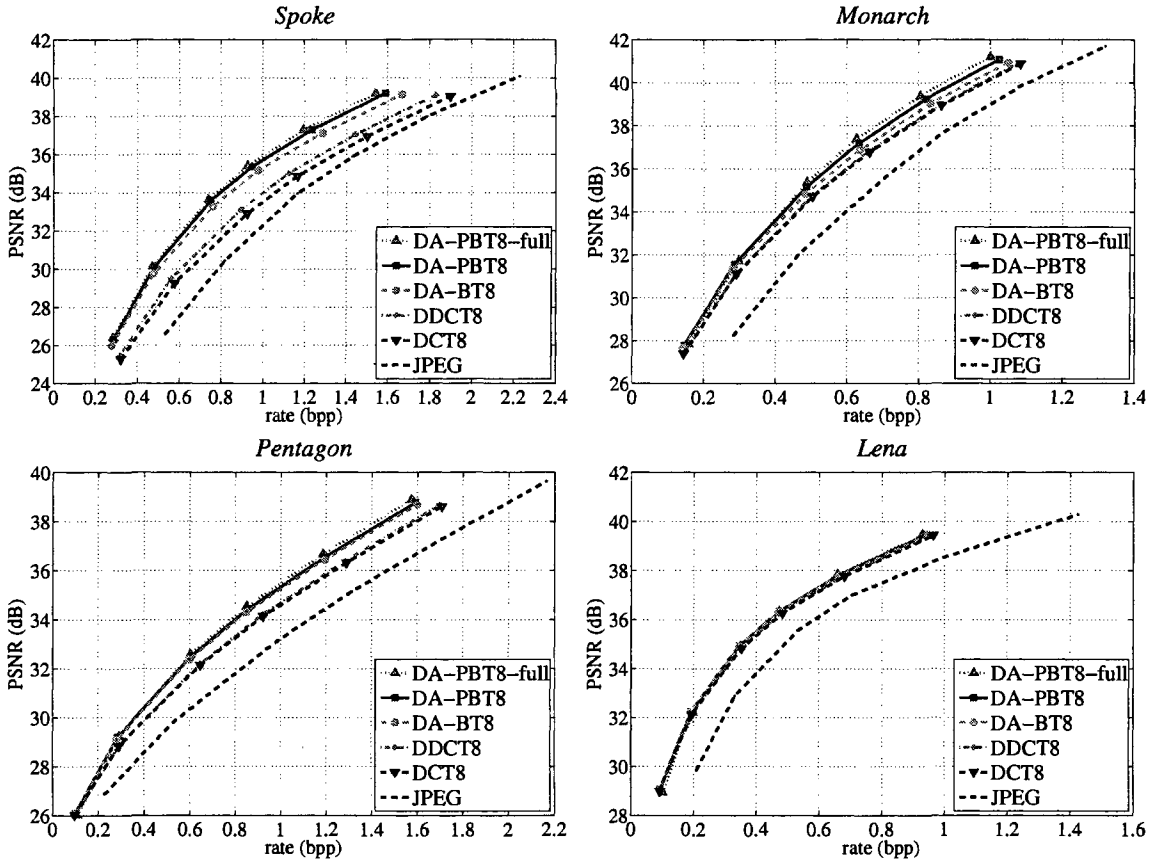


Figure 3.9: Rate-distortion performance of 8×8 transforms for image coding.

search in Sec. 3.2.3, i.e., searching through all nine modes. The rates plotted in the figure include the overhead signaling the selected blocksizes and modes.

From Fig. 3.9, it is clear that **DCT8** outperforms **JPEG** because of the more efficient entropy coder (Sec. 3.2.2). **DDCT8** only brings limited gain upon **DCT8** whereas **DA-BT8** can improve the quality over **DCT8** by more than 2 dB for images rich in directional features, mostly due to the more evenly spaced directions and the more sensible scanning orders (Sec. 3.1.1). **DA-PBT8** keeps improving the performance by spatially confining energy within partitions while inducing less complexity (Sec. 3.1.2). Finally, the gap between **DA-PBT8-full** and **DA-PBT8** indicates the performance loss due to the simplified search, which is usually negligible considering the significant reduction in complexity.

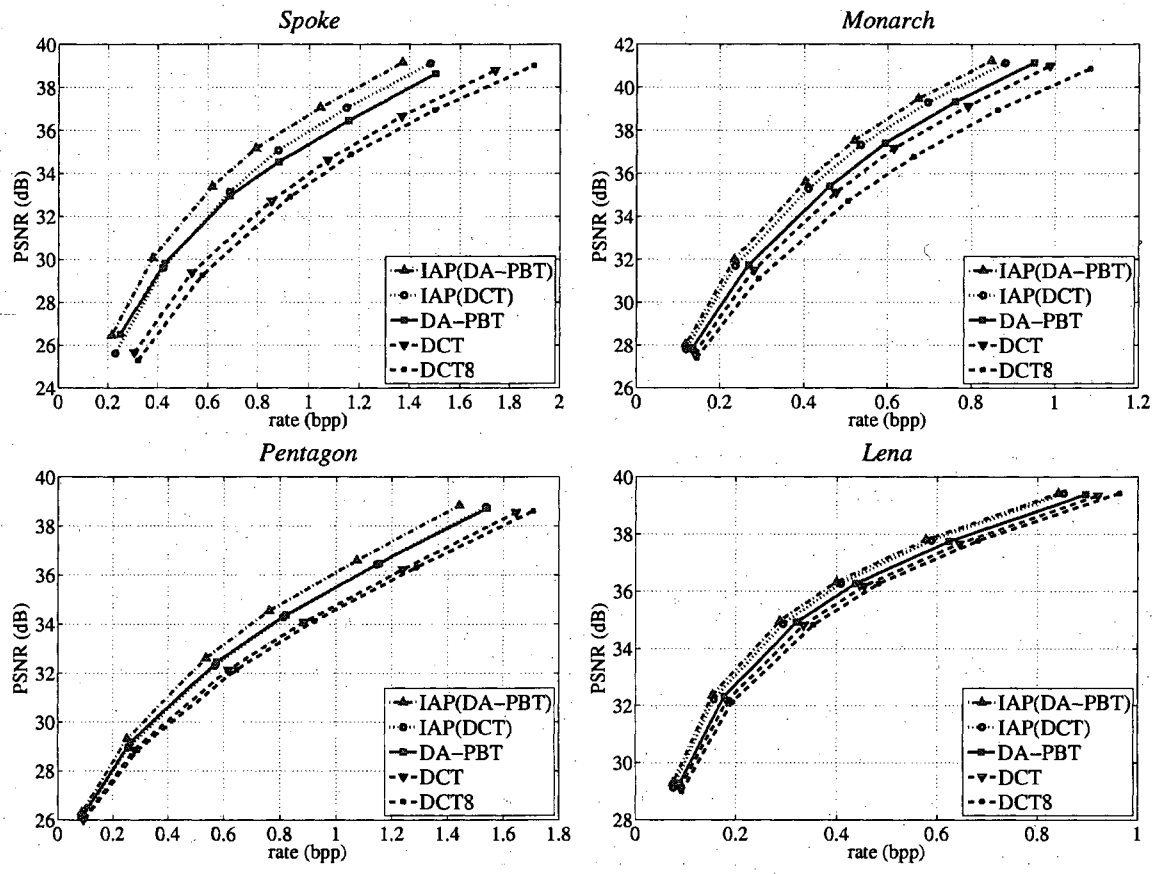


Figure 3.10: Rate-distortion performance of variable blocksize transforms for image and residual image coding.

Fig. 3.10 further includes the rate-distortion performance of variable blocksize transforms and directional intra prediction. **DCT8** again denotes the 8×8 2-D DCT, **DCT** and **DA-PBT** denote the 2-D DCT and DA-PBT with variable blocksizes (4×4, 8×8, and 16×16) respectively, and **IAP(DCT)** and **IAP(DA-PBT)** denote directional intra prediction, again with variable blocksizes, together with the 2-D DCT and the DA-PBT applied to the prediction residual respectively (Sec. 3.3.1). From Fig. 3.10, **DCT** typically provides limited gain over **DCT8** except for *Monarch* where the energy in the low-frequency content in the blurred background can be better concentrated with larger 16×16 blocks. In addition, **DA-PBT** and **IAP(DA-PBT)** outperforms **DCT** and **IAP(DCT)** respectively, and the gain generally increases with the number of sharp directional features in the image. The gap between

IAP(DA-PBT) and **IAP(DCT)** is usually smaller than that between **DA-PBT** and **DCT**. This is because in many cases the residual energy is small so that the transform coefficients are all quantized to zero regardless of which transform is applied, leading to the same rate and distortion. As an extreme example, with perfect intra prediction the residual is zero and the transform does not at all affect the coding performance. Comparing **IAP(DCT)** and **DA-PBT**, directional intra prediction is usually more efficient than the directional transforms for being able to utilize the correlation across block boundaries. However, compared to the prediction-based approach, the transform-based approach possesses two main advantages. First, with a proper design of entropy coding and bitstream organization, each macroblock (or 4×4 macroblocks when an additional 4×4 transform is applied as in the proposed approach described in Sec. 3.2.1) can be decoded independently from others, providing better support for random access and error resiliency. Second, both block-wise and image-wise, the transform is close to orthonormal, allowing embedded (quality-progressive) coding of images [170, 213].

To demonstrate the improvement in visual quality using the DA-PBT, a 256×256 region of *Pentagon* is shown in Fig. 3.13-(a), and the corresponding reconstructions from **DCT**, **DA-PBT**, **IAP(DCT)** and **IAP(DA-PBT)** using the same quantization stepsize are shown in Fig. 3.13-(b)–(e). The DA-PBT, applied both to the image and to the intra residual, greatly reduces the ringing and checkerboard artifacts around edges observed in the DCT-based schemes while demanding less rate. **DCT** with a rate comparable to that in **IAP(DA-PBT)** (Fig. 3.13(e)) is shown in Fig. 3.13(f), demonstrating the advantages of exploiting directionality in the images for image compression, both in the prediction and in the transform. Similar observations can be made for *Monarch* as shown in Fig. 3.14.

3.5.2 Video Coding

To evaluate the performance of the DA-PBT for video coding, we implemented a video coder that divides the video frames into I-pictures and P-pictures, and the 16×16 macroblocks in these pictures are categorized into three modes, INTRA, INTER, and

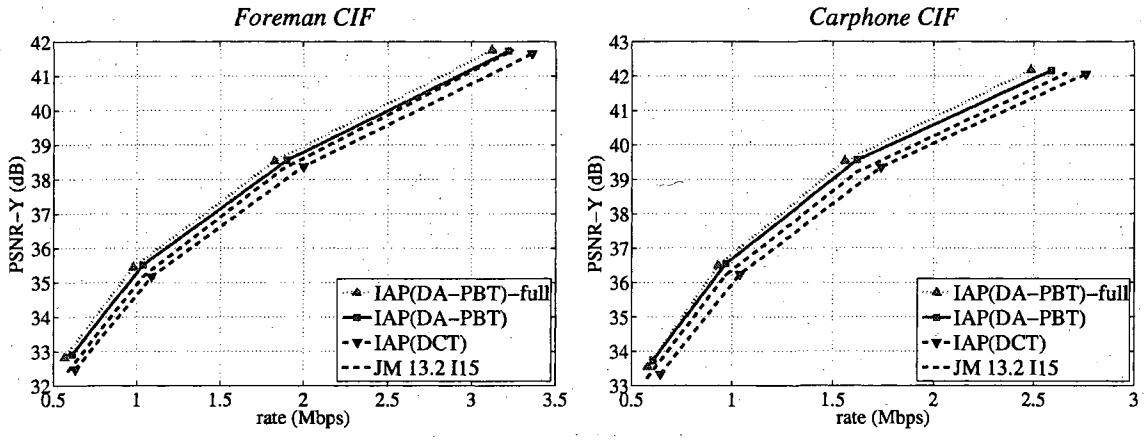


Figure 3.11: Rate-distortion performance of using the 2-D DCT and the DA-PBT for video coding with all I-pictures.

SKIP, similar to those defined in H.264 [206]. The macroblocks in the I-pictures are all in the INTRA mode and are encoded using either **IAP(DCT)** or **IAP(DA-PBT)** as described above for regular images. For the P-pictures, each macroblock can be in one of the three modes. The INTRA macroblocks are again encoded by **IAP(DCT)** or **IAP(DA-PBT)**. The INTER macroblocks are first predicted with motion-compensated inter prediction using 4×4, 8×8, or 16×16 blocks, and the residual blocks are encoded using either the 2-D DCT or the DA-PBT, denoted by **IRP(DCT)** and **IRP(DA-PBT)** respectively. A SKIP macroblock directly copies a macroblock from the reconstruction of the previous frame using motion compensation with a motion vector derived from the causal neighborhood [206]. Motion-compensated inter prediction is realized similarly to H.264 with quarter-pel accuracy and a search range of $\pm 32 \times \pm 32$ pixels. To quantize the transform coefficients, the quantization parameter $Q_{\text{H.264}}$ (Sec. 3.2.1) for the P-pictures is set to be one more than that for the I-pictures, and the parameter Δ (Sec. 3.1.3) is set to $\frac{1}{3}$ for INTRA and $\frac{1}{6}$ for INTER macroblocks [202]. The quantization indices and the overhead signaling all the mode and blocksize selections are encoded using CABAC-like entropy coding as discussed in Sec. 3.2.2, whereas the motion vectors are first predicted from causal neighborhoods and then encoded using a fixed variable-length-coding table.

We consider two coding arrangements: all I-pictures and one I-picture in every 15

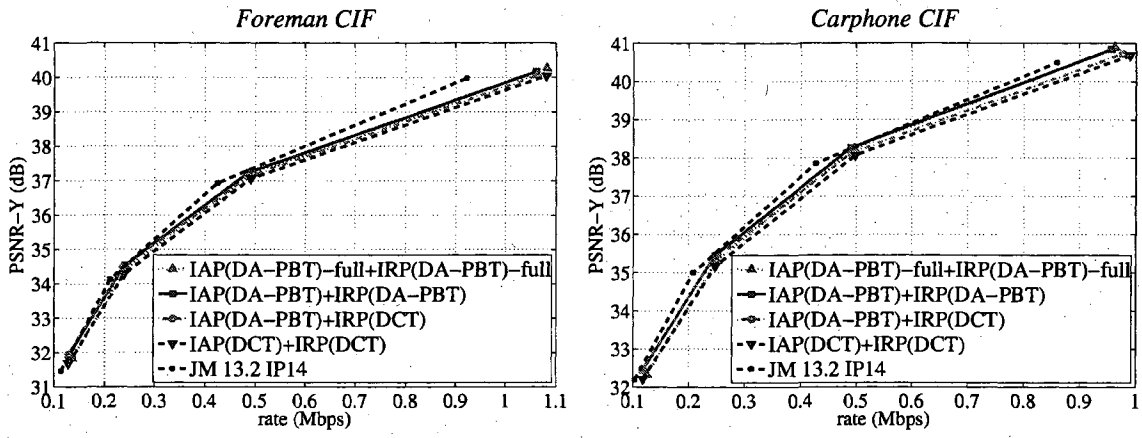


Figure 3.12: Rate-distortion performance of using the 2-D DCT and the DA-PBT for video coding with one I-picture followed by 14 P-pictures.

frames. The performance of the corresponding baseline method using the 2-D DCT is denoted by **IAP(DCT)** in Fig. 3.11 and **IAP(DCT) + IRP(DCT)** in Fig. 3.12 respectively for the CIF sequences *Foreman* and *Carphone*. To benchmark the implemented baseline methods, the rate-distortion performance of the H.264 reference software JM 13.2 is also included, denoted by **JM 13.2 I15** in Fig. 3.11 and **JM 13.2 IP14** in Fig. 3.12 for the two arrangements respectively, using the High Profile (FREXt) that additionally enables 8×8 intra prediction and transforms [111, 141]. From Fig. 3.12, at high rates **IAP(DCT) + IRP(DCT)** is evidently less efficient than **JM 13.2 IP14** due to several simplifications in our implementation. For instance, the High Profile in H.264 allows the 4×4 or the 8×8 transform to be selected adaptively in INTER macroblocks [141], whereas in our implementation the size of the transform is coupled to the blocksize used in motion-compensated prediction. Nonetheless, the implemented coder in general delivers performance comparable to the state-of-the-art video coding standard.

For using all I-pictures, compared to **IAP(DCT)**, **IAP(DA-PBT)** on average improves the reconstruction quality by 0.51 dB for *Foreman* and 0.66 dB for *Carphone*, and, equivalently, reduces the rate by 9.0% and 10.5% respectively as shown in Fig. 3.11. Furthermore, instead of using the same mode for the DA-PBT and directional intra prediction, the performance of considering all of the nine DA-PBT

modes regardless the intra prediction mode selected is also included in the figure, denoted by **IAP(DA-PBT)-full**. The performance loss from using the same mode is generally acceptable considering the additional computation required to evaluate the rate and distortion associated with all the other DA-PBT modes.

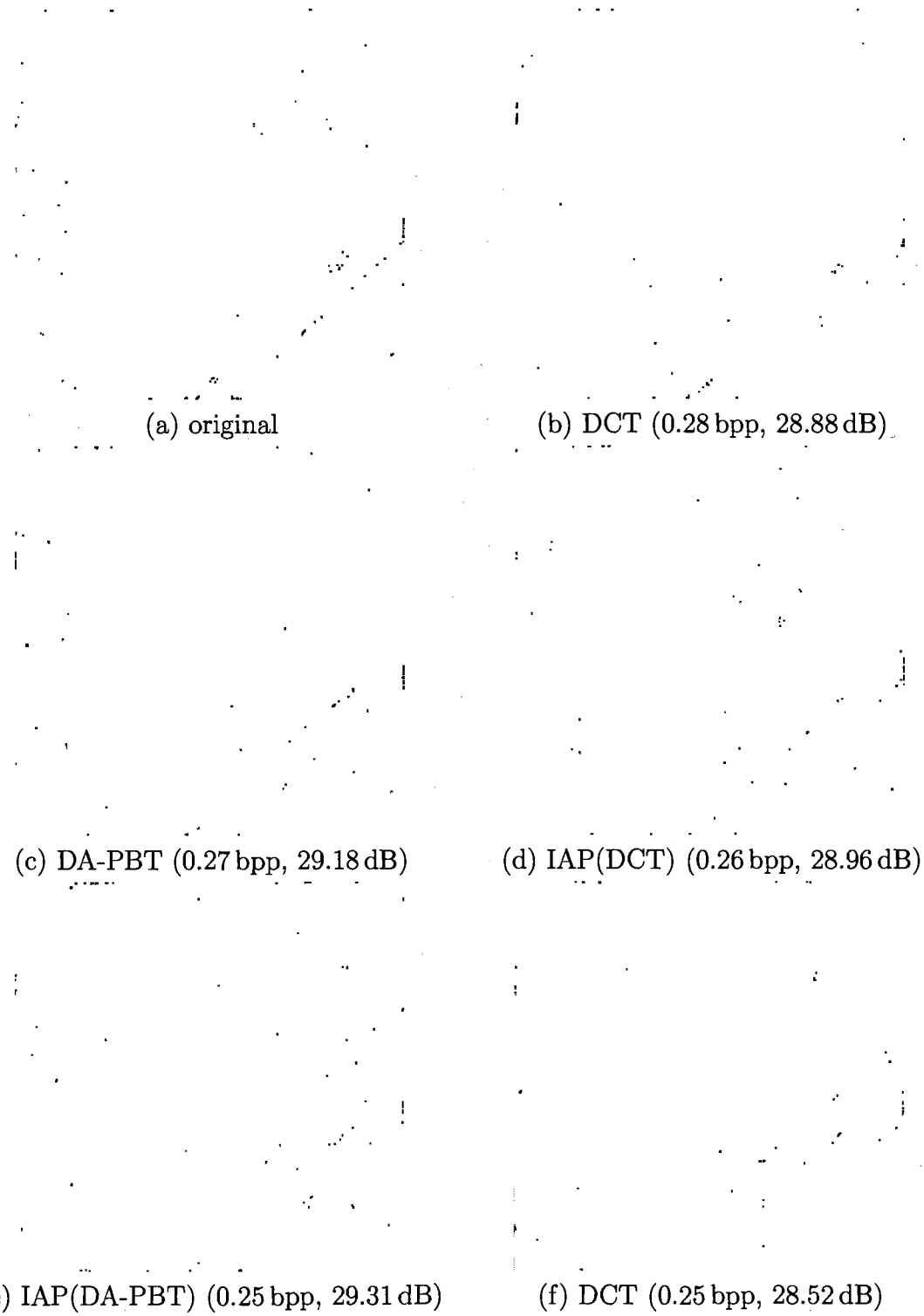
For the arrangement with inter prediction, the DA-PBT can be applied to the INTRA macroblocks, denoted by **IAP(DA-PBT) + IRP(DCT)**, or additionally to the INTER macroblocks, denoted by **IAP(DA-PBT) + IRP(DA-PBT)** in Fig. 3.12. As discussed in Sec. 3.3.2, **IRP(DA-PBT)** considers at most one directional mode of the DA-PBT. The scheme that enables all nine modes of the DA-PBT to be selected for both the intra and the inter residual blocks is denoted by **IAP(DA-PBT)-full + IRP(DA-PBT)-full** and is also included in the figure. Although in Fig. 3.11 **IAP(DA-PBT)-full** outperforms **IAP(DA-PBT)**, considering all DA-PBT modes for the inter residual as in **IAP(DA-PBT)-full + IRP(DA-PBT)-full** surprisingly deteriorates the performance. Despite the additional flexibility, **IAP(DA-PBT)-full + IRP(DA-PBT)-full** no longer achieves better rate-distortion performance than **IAP(DA-PBT) + IRP(DA-PBT)** because of the increased overhead for the INTER macroblocks that offsets the potential gain (Sec. 3.3.2). On average, compared to **IAP(DCT) + IRP(DCT)**, **IAP(DA-PBT) + IRP(DA-PBT)** improves the quality by 0.22 dB for *Foreman* and 0.27 dB for *Carphone*, and equivalently reduces the rate by 5.5% and 6.6% respectively. Similar to comparing the effectiveness of the DA-PBT on image blocks and intra residual blocks, the DA-PBT is less effective on inter than intra residual since inter prediction is typically more accurate and leads to smaller residual energy. This contradicts the observation in [219] where the DDCT is more effective on inter residual blocks rather than on image blocks. We conjecture that this is due to the less accurate motion compensation adopted in [219] that always uses 16×16 blocks and a search range of mere $\pm 7 \times \pm 7$ pixels.

3.6 Summary

We have proposed a new direction-adaptive partitioned block transform (DA-PBT) for coding of images and video sequences. The DA-PBT outperforms the conventional

2-D DCT by more than 2 dB for images with pronounced directional features. Since it avoids the typical ringing and checkerboard artifacts of the 2-D DCT, subjective improvements are even larger than indicated by the rate-distortion performance. The DA-PBT also outperforms a previously proposed directional DCT while demanding less computation, owing to an improved directional selectivity, direction-adaptive partitioning and better coefficient ordering for entropy coding. As for conventional block transforms, the DA-PBT achieves its best performance when combined with adaptive blocksizes. To exploit the frequency response of the human visual system, quantization matrices can be used with the DA-PBT. We show a straightforward way to transform JPEG quantization matrices into the directional transform space, thus avoiding cumbersome subjective tests.

We have also explored the combination of the DA-PBT with predictive coding, either directional prediction within the image or interframe prediction for video. For intraframe directional prediction, the direction selection for the prediction and the transform can be elegantly combined. It is not surprising that the gains of both techniques are not additive, as similar signal properties are exploited by the prediction and the transform. Since the DA-PBT operates in a block-wise manner, the incorporation into block-based motion-compensated video coding is straightforward. Alas, we have not been able to demonstrate significant gains by compressing the motion-compensated prediction residual with the DA-PBT when advanced motion compensation is used for our relatively simple test sequences. The DA-PBT might still have a role to play for video coding, at least for intra-coded blocks, or where motion-compensated prediction is not fully effective.

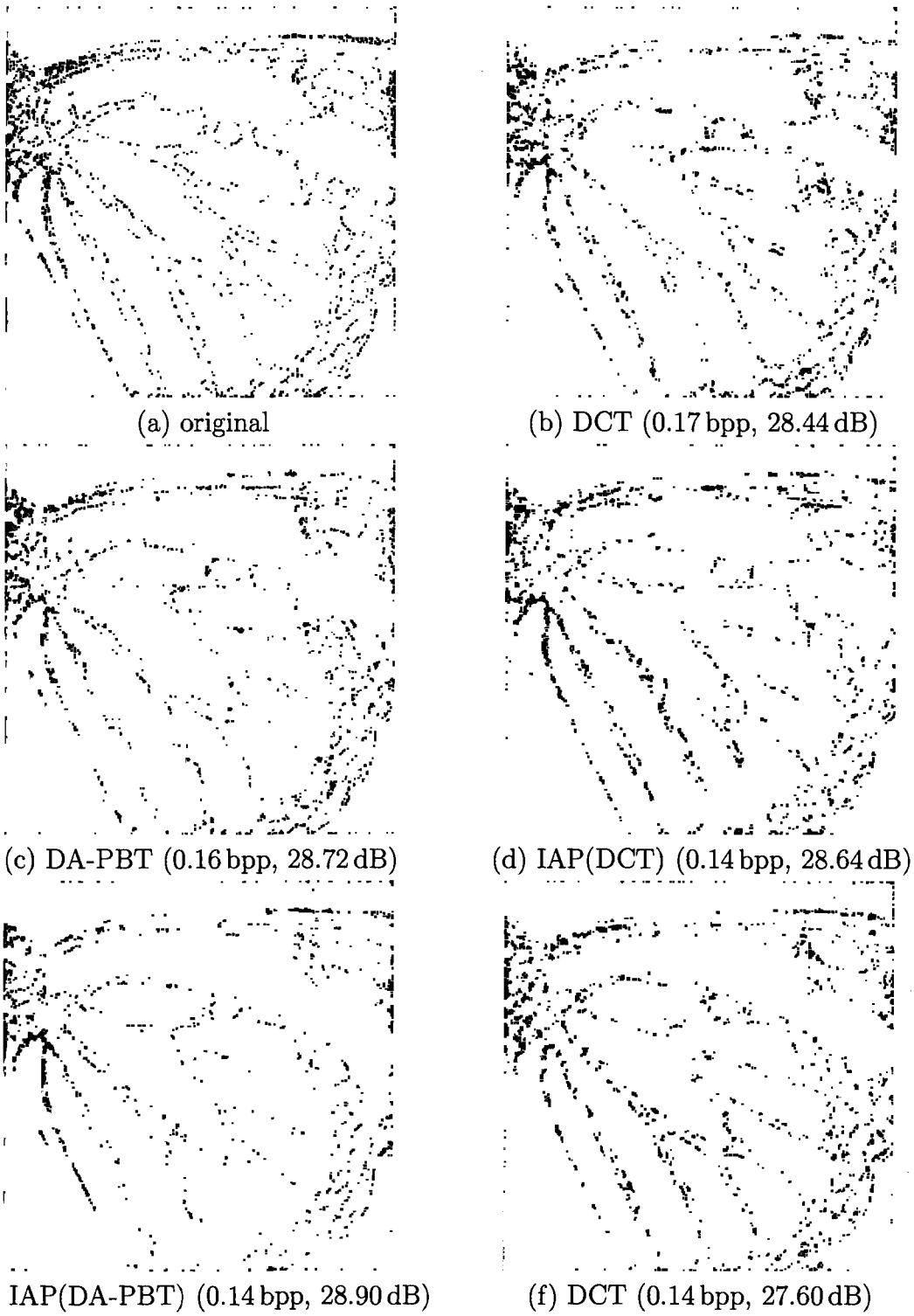


(a) original (b) DCT (0.28 bpp, 28.88 dB)

(c) DA-PBT (0.27 bpp, 29.18 dB) (d) IAP(DCT) (0.26 bpp, 28.96 dB)

(e) IAP(DA-PBT) (0.25 bpp, 29.31 dB) (f) DCT (0.25 bpp, 28.52 dB)

Figure 3.13: Reconstruction of a 256×256 region in the *Pentagon* image.



(a) original (b) DCT (0.17 bpp, 28.44 dB)

(c) DA-PBT (0.16 bpp, 28.72 dB) (d) IAP(DCT) (0.14 bpp, 28.64 dB)

(e) IAP(DA-PBT) (0.14 bpp, 28.90 dB) (f) DCT (0.14 bpp, 27.60 dB)

Figure 3.14: Reconstruction of a 256 × 256 region in the *Monarch* image.

Chapter 4

Direction-Adaptive Discrete Wavelet Transform

In the previous chapter, we have shown that the DA-PBT, as a block transform, can be easily combined with block-based predictive coding widely adopted in image and video coding techniques. However, due to the block-wise operation, the transform is unable to exploit the correlation across blocks. Moreover, at lower rates, the reconstruction exhibits severe blocking artifacts, requiring additional deblocking filtering to partially mitigate them. Alternatively, image coding with the wavelet transform applied over the entire image is free of blocking artifacts and often leads to better performance both in rate-distortion measurements and in visual quality. In this chapter, we propose a direction-adaptive transform based on the wavelet transform, referred to as the direction-adaptive discrete wavelet transform (DA-DWT). For image coding, the best mode of the DA-DWT is selected at the encoder for each image block using a rate-distortion optimized framework and signaled to the decoder as side information.

In Sec. 4.1, we describe how different modes of the DA-DWT are constructed using directional lifting. A practical image coding framework using the DA-DWT is discussed in Sec. 4.2. Theoretical analysis of the DA-DWT and performance evaluation using the image model proposed in Chap. 2 are presented in Sec. 4.3. Finally, experimental results demonstrating the superior performance of the DA-DWT both objectively and subjectively are included in Sec. 4.4.

4.1 Construction of DA-DWT

4.1.1 2-D DWT with Lifting

The conventional 2-D DWT in general consists of two stages. In Stage 1, the analysis filters of the 1-D DWT are applied to the image vertically followed by vertical subsampling to obtain the low-pass subband, L, and the high-pass subband, H. In the second stage, the analysis filters are again applied to L and H horizontally, followed by horizontal subsampling to obtain the LL and LH in Stage 2-L, and the HL and HH subband in Stage 2-H respectively. It is shown in [179] that any two-band biorthogonal DWT can be factored into pairs of lifting steps. We limit the discussion to wavelet transforms that can be realized with one pair of the lifting steps, i.e., one prediction step followed by one update step. For instance, the Haar wavelet and the family of interpolating wavelets all belong to this category [16, 47].

Let $\mathbf{s} = \{s[\mathbf{l}] \mid \mathbf{l} \in \mathbf{\Pi}\}$, where $\mathbf{l} = (l_x, l_y)^T$ and $s[\mathbf{l}] = s[l_x, l_y]$, denote a set of image samples on a 2-D orthogonal sampling grid $\mathbf{\Pi} = \{(l_x, l_y)^T \in \mathbb{Z}^2\}$. The grid $\mathbf{\Pi}$ is composed of 4 sub-grids: $\mathbf{\Pi}_{pq} = \{(l_x, l_y)^T \in \mathbf{\Pi} \mid l_y \bmod 2 = p,\ l_x \bmod 2 = q\}$. To apply the 2-D DWT with lifting, we first apply a Stage-1 transform between the even and the odd rows of the image, i.e., between $\mathbf{s}_0 = \{s[\mathbf{l}_0] \mid \mathbf{l}_0 \in \mathbf{\Pi}_0 = \mathbf{\Pi}_{00} \cup \mathbf{\Pi}_{01}\}$ and $\mathbf{s}_1 = \{s[\mathbf{l}_1] \mid \mathbf{l}_1 \in \mathbf{\Pi}_1 = \mathbf{\Pi}_{10} \cup \mathbf{\Pi}_{11}\}$. Denote the resulting low-pass subband by $\mathbf{w}_0 = \{w_0[\mathbf{l}_0] \mid \mathbf{l}_0 \in \mathbf{\Pi}_0\}$ and the high-pass subband by $\mathbf{w}_1 = \{w_1[\mathbf{l}_1] \mid \mathbf{l}_1 \in \mathbf{\Pi}_1\}$, the lifting steps can generally be expressed as

$$w_1[\mathbf{l}_1] = g_H \cdot (s[\mathbf{l}_1] - P_{\mathbf{l}_1}(\mathbf{s}_0)),\ \ \forall\ \mathbf{l}_1 \in \mathbf{\Pi}_1, \tag{4.1a}$$

$$w_0[\mathbf{l}_0] = g_L \cdot (s[\mathbf{l}_0] + g_H^{-1} \cdot U_{\mathbf{l}_0}(\mathbf{w}_1)),\ \ \forall\ \mathbf{l}_0 \in \mathbf{\Pi}_0, \tag{4.1b}$$

where the prediction function, $P_{\mathbf{l}_1}(\cdot)$, and the update function, $U_{\mathbf{l}_0}(\cdot)$, are functions of the sample values in the input with a scalar output, and g_L and g_H are scaling factors. The Stage-1 transform is realized by first performing the prediction step in (4.1a) followed by the update step in (4.1b) so that $\mathbf{s}$ is decomposed into $\mathbf{w}_0$ and $\mathbf{w}_1$. To reconstruct $\mathbf{s}$ from $\mathbf{w}_0$ and $\mathbf{w}_1$, the inverse transform can also be realized by the

following lifting steps:

$$s[\mathbf{l}_0] = g_L^{-1} \cdot w_0[\mathbf{l}_0] - g_H^{-1} \cdot U_{\mathbf{l}_0}(\mathbf{w}_1), \quad \forall \ \mathbf{l}_0 \in \mathbf{\Pi}_0, \tag{4.2a}$$

$$s[\mathbf{l}_1] = g_H^{-1} \cdot w_1[\mathbf{l}_1] + P_{\mathbf{l}_1}(\mathbf{s}_0), \quad \forall \ \mathbf{l}_1 \in \mathbf{\Pi}_1. \tag{4.2b}$$

Using the lifting steps, the prediction function $P_{\mathbf{l}_1}(\cdot)$ and the update function $U_{\mathbf{l}_0}(\cdot)$ can be space-varying or even nonlinear without affecting the reversibility of the transform as shown in (4.1) and (4.2). This property has been utilized to design locally adaptive transforms for image coding [13, 26, 30, 40, 53, 79, 80, 182, 201], video coding [133, 152, 168], and light field compression [31, 32, 83].

Similarly, the transform between the even and the odd columns of the samples further decomposes $\mathbf{w}_0$ in Stage 2-L into $\mathbf{w}_{00}$ and $\mathbf{w}_{01}$, defined on $\mathbf{\Pi}_{00}$ and $\mathbf{\Pi}_{01}$ respectively, and decomposes $\mathbf{w}_1$ in Stage 2-H into $\mathbf{w}_{10}$ and $\mathbf{w}_{11}$, defined on $\mathbf{\Pi}_{10}$ and $\mathbf{\Pi}_{11}$ respectively, where $\mathbf{w}_{00}$, $\mathbf{w}_{01}$, $\mathbf{w}_{10}$, and $\mathbf{w}_{11}$ correspond to the LL, LH, HL, and HH subband of the image respectively. We refer to this process of decomposing the image samples $\mathbf{s}$ into the 4 subbands as one level of the 2-D DWT. Multiple levels of the transform may be performed by iteratively applying this process to the resulting LL subband.

4.1.2 Directional Lifting

In the conventional 2-D DWT, the prediction and the update function in (4.1) can be expressed as

$$P_{\mathbf{l}_1}(\mathbf{s}_0) = \sum_{k=-K_P}^{K_P-1} c_{P,k} \cdot s[l_{1,x}, l_{1,y} - (2k+1)], \tag{4.3a}$$

$$U_{\mathbf{l}_0}(\mathbf{w}_1) = \sum_{k=-K_U}^{K_U-1} c_{U,k} \cdot w_1[l_{0,x}, l_{0,y} - (2k+1)], \tag{4.3b}$$

where K_P, $c_{P,k}$, K_U, and $c_{U,k}$ are determined by the wavelet kernel adopted. For instance, for the popular 5/3 filter pair [42, 77], $K_P = 1$ with $c_{P,-1} = c_{P,0} = \frac{1}{2}$, $K_U = 1$ with $c_{U,-1} = c_{U,0} = \frac{1}{4}$, and g_L and g_H are customarily set to $\sqrt{2}$ and $\frac{1}{\sqrt{2}}$

respectively [47,179]. Note that only samples in the same column are involved in the transform.

For image compression, for each sample in the high-pass subband, $w_1[\mathbf{l}_1]$, it is generally desirable to select a prediction function $P_{\mathbf{l}_1}(\mathbf{s}_0)$ in (4.1a) that predicts $s[\mathbf{l}_1]$ from the samples in $\mathbf{s}_0$ such that the energy in the residual $w_1[\mathbf{l}_1]$ is minimized and most energy is concentrated in the corresponding low-pass subband $w_0[\mathbf{l}_0]$. In the proposed DA-DWT, we define directional prediction filters with direction $\mathbf{d} = (d_x, d_y)^T$ from which $P_{\mathbf{l}_1}(\mathbf{s}_0)$ can be adaptively selected as

$$P^{\mathbf{d}}_{\mathbf{l}_1}(\mathbf{s}_0) = \sum_{k=-K_P}^{K_P-1} c_{P,k} \cdot s[\mathbf{l}_1 - (2k+1)\mathbf{d}], \tag{4.4}$$

where $\mathbf{d}$ is defined such that

$$\mathbf{l}_1 - (2k+1)\mathbf{d} \in \mathbf{\Pi}_0, \ \ \forall \ \mathbf{l}_1 \in \mathbf{\Pi}_1, \ k = -K_P, \cdots, K_P - 1. \tag{4.5}$$

The directional prediction filter corresponds to performing the prediction step along a Stage-1 direction $\mathbf{d}$. From (4.5), $\mathbf{d} \in \mathbb{Z}^2$ and d_y is always odd. We further constrain $\mathbf{d}$ such that the line segment from $(0,0)^T$ to $(d_x, d_y)^T$ does not intersect with any other point in $\mathbf{\Pi}$, i.e., d_x and d_y are coprime integers. For instance, the following directions are used in our previous work in [26]: $(-3,1)^T$, $(-2,1)^T$, $(-1,1)^T$, $(-1,3)^T$, $(0,1)^T$, $(1,3)^T$, $(1,1)^T$, $(2,1)^T$, and $(3,1)^T$.

Denote the direction selected at location $\mathbf{l}_1$ for $P_{\mathbf{l}_1}(\mathbf{s}_0)$ as $\mathbf{d}^\star_{\mathbf{l}_1}$. Upon completion of the prediction step of all samples, the corresponding update function in (4.1b) is defined as

$$U_{\mathbf{l}_0}(\mathbf{w}_1) = \sum_{k=-K_U}^{K_U-1} c_{U,k} \cdot \sum_{\{\mathbf{l}_1 \mid \mathbf{l}_1 - (2k+1)\mathbf{d}^\star_{\mathbf{l}_1} = \mathbf{l}_0\}} w_1[\mathbf{l}_1]. \tag{4.6}$$

In words, wherever an image sample $s[\mathbf{l}_1]$ is predicted by $c_{P,k} \cdot s[\mathbf{l}_0]$, $s[\mathbf{l}_0]$ is updated by $c_{U,k} \cdot w_1[\mathbf{l}_1]$. Note that if direction $(0,1)^T$ is always selected, i.e., $\mathbf{d}^\star_{\mathbf{l}_1} = (0,1)^T \ \forall \ \mathbf{l}_1$, (4.4) and (4.6) are equivalent to the functions in (4.3) and directional lifting is identical to

the conventional DWT.

For the second stage of the transform further applied to $\mathbf{w}_0$ and $\mathbf{w}_1$, the transform is applied along a Stage-2 direction, $\check{\mathbf{d}} = (\check{d}_x, \check{d}_y)^T$, where $\check{d}_x$ and $\check{d}_y$ are again coprime integers, and $\check{d}_x$ is odd and $\check{d}_y$ is even. Consequently,

$$\mathbf{l}_{01} - (2k+1)\check{\mathbf{d}} \in \mathbf{\Pi}_{00}, \;\; \mathbf{l}_{11} - (2k+1)\check{\mathbf{d}} \in \mathbf{\Pi}_{10}, \qquad (4.7)$$
$$\forall \;\; \mathbf{l}_{01} \in \mathbf{\Pi}_{01}, \; \mathbf{l}_{11} \in \mathbf{\Pi}_{11}, \; k = -K_P, \cdots, K_P - 1.$$

The resulting candidates for the prediction function and the corresponding update function are defined similarly as in (4.4) and (4.6). Note that the direction selected for Stage 2-L and Stage 2-H can be different in general.

Several other approaches in the literature that also enable directional adaptation through lifting essentially adopt a set of sub-pel directions achieved by spatial interpolation, e.g., $\mathbf{d} = \{(\frac{i}{K}, 1)^T, \;\; i = -K, \cdots, K\}$, where K is typically 2 or 4 [53,182,201]. These approaches assume $d_y = 1$ (and $\check{d}_x = 1$) whereas the proposed DA-DWT framework only requires d_y (and $\check{d}_x$) to be odd. In the experimental results as well as the mathematical analysis reported in [26], we have shown that using integer samples, e.g., $(-1, 3)^T$, directly for prediction is typically more efficient than using the interpolated ones, e.g., $(-\frac{1}{2}, 1)^T$, in the presence of sharp image features. Additionally, the integer directions do not involve sub-pel interpolation, and hence require less computation.

Furthermore, in the sub-pel approaches only the directions between $\pm 45°$ from the vertical axis are considered in Stage 1. For horizontal-ish image features, there is no closely aligned filtering directions and thus the energy is spread into the high-pass subband H. As a result, for image compression they favor vertical-ish image features, and therefore are sensitive to image transposition. In our previous work [30], quincunx subsampling was adopted to provide directions covering the 2-D space more evenly and has shown reduced sensitivity to image transposition. However, DWT coding using quincunx subsampling is less efficient for typical images [87]. In the proposed DA-DWT, we retain the conventional orthogonal subsampling as described in Sec. 4.1.1 and allow integer directions beyond $\pm 45°$, e.g., $(-2, 1)^T$, to provide a

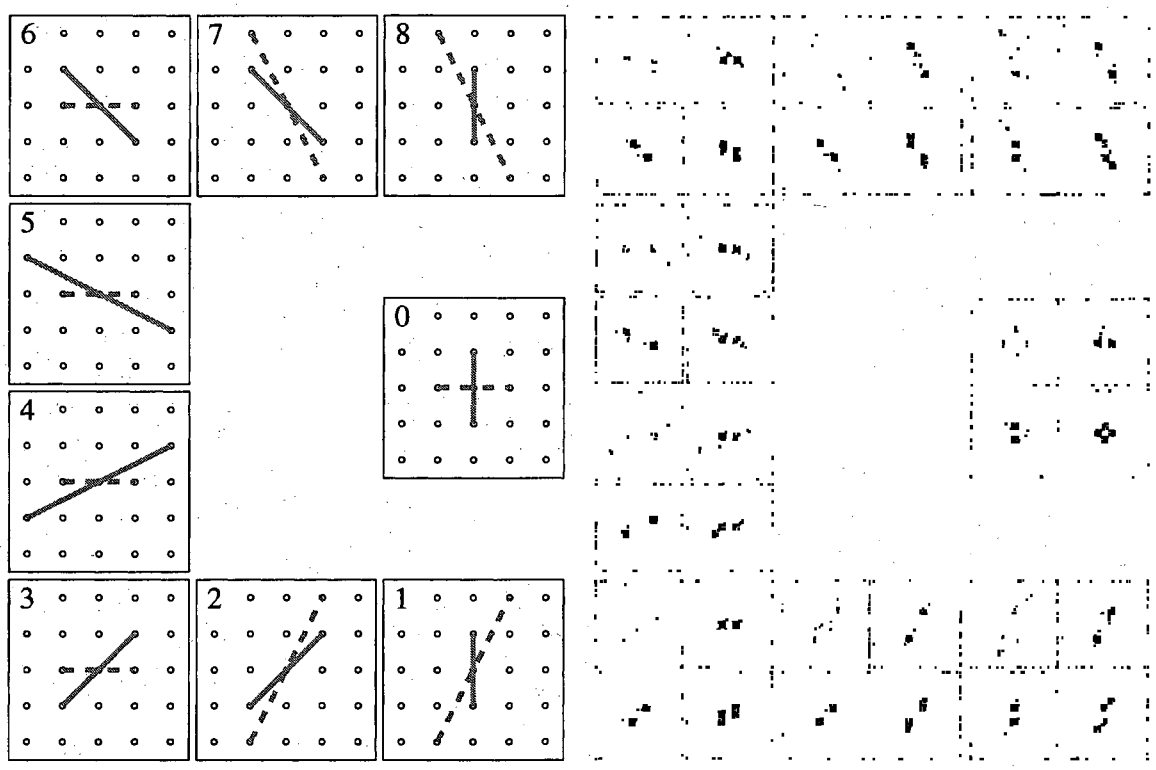
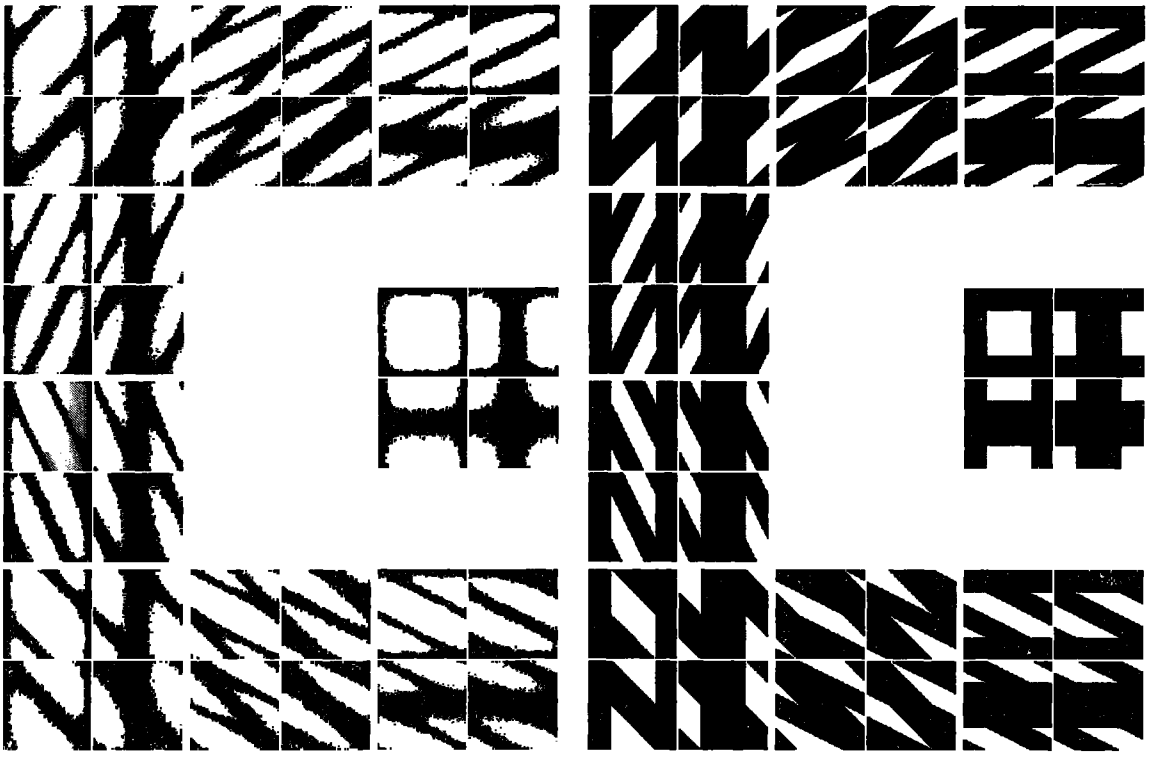


Figure 4.1: (a) The direction-pairs in the 9 modes of the DA-DWT. The solid lines denote the Stage-1 directions $\mathbf{d}$, and the dashed lines denote the Stage-2 directions $\check{\mathbf{d}}$. (b) The impulse responses of the analysis filters of one level of the DA-DWT using the 5/3 filter pair. In each mode, the four plots correspond to the (top-left) LL, (top-right) LH, (bottom-left) HL, and (bottom-right) HH subband.

larger directional span, instead of using the limited set of sub-pel directions. Similar to the case with quincunx subsampling, reduced sensitivity to transposition is observed. The mathematical analysis in [26] also agrees with this observation.

4.1.3 Direction Pairing

To apply the DA-DWT to an image region, in general three directions need to be determined in Stage 1, Stage 2-L, and Stage 2-H of the transform as described in our previous work [26]. In practice, these directions are usually aligned with the image features so that the energy in the high-pass subband is minimized. Consequently,



(a) (b)

Figure 4.2: The frequency responses of the 2-D analysis filters of one level of the DA-DWT using (a) the 5/3 filter pair, and (b) the sinc wavelet. In each mode, the four plots correspond to the (top-left) LL, (top-right) LH, (bottom-left) HL, and (bottom-right) HH subband. Each plot represents the 2-D discrete-space frequency $-\pi \leq \Omega_x \leq \pi$ from left to right and $-\pi \leq \Omega_y \leq \pi$ from bottom to top.

corresponding Stage-1 and Stage-2 directions are usually selected. For instance, the pair of $\mathbf{d} = (0,1)^T$ for Stage 1 and $\check{\mathbf{d}} = (-1,2)^T$ for Stage 2-L and 2-H is likely to be selected if vertical-ish features appear in the region.

We identify 8 such pairs that cover a variety of feature orientations, and they constitute the 8 directional modes of the DA-DWT shown in Fig. 4.1-(a) (Mode 1~8) together with Mode 0 that falls back to the conventional 2-D DWT. Different modes are adaptively selected according to the image features contained in local image regions. Additionally, the impulse response of the resulting 2-D analysis filters of one level of the DA-DWT using the 5/3 filter pair are shown in Fig. 4.1-(b). Different from the filters in the conventional 2-D DWT (Mode 0), the directional modes use

elongated and oriented 2-D filters that are realized by cascading simple 1-D lifting steps. The corresponding frequency responses of the DA-DWT using the 5/3 filter pair are shown in Fig. 4.2-(a). If a hypothetical sinc wavelet that requires an infinite support is used, i.e., $K_P = \infty$ with $c_{P,k} = \text{sinc}(\frac{2k+1}{2})$, $K_U = \infty$ with $c_{U,k} = \frac{1}{2}c_{P,k}$, and $g_L = \sqrt{2}$ and $g_H = \frac{1}{\sqrt{2}}$ in (4.3), the resulting frequency responses are shown in Fig. 4.2-(b). Using the sinc wavelet, the 2-D frequency space is partitioned by the DA-DWT into 4 directional subbands. With the practical 5/3 filter pair, the subbands overlap but still retain the desired directional supports.

Note that we only include directions with a maximal displacement of 2 in each dimension, i.e., $|d_x|$, $|d_y|$, $|\check{d}_x|$, $|\check{d}_y| \leq 2$. This ensures that the size of the filter support does not increase much from the conventional 2-D DWT in order to reduce the ringing artifacts at low rates. The number of modes can be increased to handle more feature orientations. Nonetheless, through experiments we found that using more than the proposed eight directional modes provides only limited gain for typical images.

4.2 Image Coding with DA-DWT

4.2.1 Direction Selection

For image coding, the DA-DWT modes selected at the encoder need to be signaled to the decoder so that the transform can be properly reversed to reconstruct the image samples. The modes should be selected to minimize the distortion of the reconstructed image for a given rate budget. The rate budget is spent on the overhead required to signal the selection and the rate for coding the wavelet coefficients. To reduce the amount of the signaling overhead, the modes are selected in a block-wise fashion. Finding the best DA-DWT modes is therefore analogous to rate-constrained block-based motion estimation in video coding [82, 204, 205].

Assume L levels of the DA-DWT are to be applied to an image to decompose the image into $3L + 1$ subbands. We divide the image into macroblocks of $S_B \times S_B$ pixels. Each macroblock may contain a single $S_B \times S_B$ block, or be further divided into blocks of $\frac{S_B}{2} \times S_B$, $S_B \times \frac{S_B}{2}$, $\frac{S_B}{2} \times \frac{S_B}{2}$, $\frac{S_B}{4} \times S_B$, $S_B \times \frac{S_B}{4}$, $\frac{S_B}{4} \times \frac{S_B}{2}$, $\frac{S_B}{2} \times \frac{S_B}{4}$, or

$\frac{S_B}{4} \times \frac{S_B}{4}$ pixels. Every block is assigned one of the nine modes of the DA-DWT.

Ideally, for each macroblock the best blocksize and the mode in each block are selected as the blocksize and mode combination that minimizes a Lagrangian cost function $\bar{D}_c + \bar{\lambda}(\bar{R}_c + \bar{R}_s)$, the same as for the DA-PBT discussed in Sec. 3.2.1. In this cost function, $\bar{D}_c$ denotes the distortion (sum of squared error) in the reconstructed macroblock, $\bar{R}_c$ and $\bar{R}_s$ denote the number of bits required to encode the transform coefficients and the overhead signaling the selection respectively, and $\bar{\lambda}$ is the Lagrangian multiplier set to $0.85 \cdot 2^{(Q_{\text{H.264}}-12)/3}$ where $Q_{\text{H.264}}$ depends on the target quality and can be mapped into a quantization stepsize as defined in H.264 [206].

Different from the DA-PBT, the DA-DWT (or DWT in general) is not a block transform and filtering in the prediction and the update step extends across block boundaries. Therefore, the transform coefficients in the current block depends not only on the mode selected for the block, but also on the modes in all of the neighboring blocks, making it impossible to evaluate the distortion and rate without knowing the modes in the subsequent blocks. Furthermore, entropy coders designed for non-block-transforms such as the DWT may not necessarily work in a block-by-block manner. Evaluating the rate for encoding the coefficients in a block can thus be difficult. As a simplification, we first approximate the transform coefficients in the current block for a certain mode by applying the DA-DWT assuming that the same mode is selected in all the involved neighboring blocks. Additionally, instead of using $\bar{D}_c + \bar{\lambda}(\bar{R}_c + \bar{R}_s)$, the cost function $\bar{S}_c + \sqrt{\bar{\lambda}}\bar{R}_s$, where $\bar{S}_c$ is the sum of absolute values of all the wavelet coefficients in the macroblock, is adopted as it has been shown to be an effective alternative in rate-constrained motion estimation [204, 205].

Once the blocksize and the corresponding modes have been selected for every macroblock, the actual transform coefficients of the entire image are generated by first performing the prediction step in the Stage-1 transform as described in (4.1a) and (4.4) with the selected locally varying directions. Upon completion of the prediction step for all the samples, the update step in the Stage-1 transform is performed as in (4.1b) and (4.6), followed by the lifting steps in the Stage-2 transforms. The process is iteratively applied to the resulting low-resolution LL subband to achieve multiple levels of the transform. Note that because filtering extends across block boundaries,

unlike the DA-PBT, blocking artifacts are usually absent in the reconstructed images and therefore deblocking filters are not needed.

In the above discussion, we consider the case where the L levels of the DA-DWT share the same blocksize and mode selection. In general, at different resolution levels of the transform, the macroblock-size and the selected blocksizes and modes can all be different, requiring a separate selection procedure at each level of the transform as in [26, 53]. This may be beneficial for high-frequency periodic patterns in the image such as fine stripes, since at different levels the patterns may appear having different orientations due to the aliasing from downsampling. Nevertheless, we observed that for most image features it is sufficient to select the same mode across resolution levels because the feature orientations are usually resolution-invariant.

At block boundaries, using the Stage-1 transform for example, a sample at $\mathbf{l}_0 \in \mathbf{\Pi}_0$ may be used to predict samples in $\mathbf{\Pi}_1$ in more than one directions. This happens when, for instance, $n(\{\mathbf{l}_1 \mid \mathbf{l}_1 + \mathbf{d}^{\star}_{\mathbf{l}_1} = \mathbf{l}_0\}) > 1$, where $n(\mathcal{A})$ denotes the cardinality of set $\mathcal{A}$, analogous to the situation of multiply-connected pixels in video coding with motion-compensated temporal filtering [145]. Meticulous rules have been developed to cope with this situation in order to make adaptive transforms reversible, both for video coding [145] and image coding [151, 200]. Owing to the lifting structure, the proposed DA-DWT does not require these rules and reversibility of the transform is always ensured as shown in (4.1) and (4.2).

The situation of multiply-connected pixels also incurs ambiguity in the direction of the update step. The update functions we propose in (4.6), analogous to the barbell lifting scheme for video coding [210, 211], has been shown to achieve better compression performance than the update functions in [52, 53] and [30], analogous to using the reversed motion vectors in the update step [210]. The discussion in Sec. 4.1.2 is limited to wavelet kernels that can be realized using one pair of the lifting steps. For kernels that can only be constructed with multiple pairs of lifting steps, such as the 9/7 filter pair [47, 179], a more complicated update scheme is required to improve rate-distortion efficiency and reduce coding artifacts around block boundaries, as studied in [129].

All the aforementioned update schemes applied at block boundaries are based

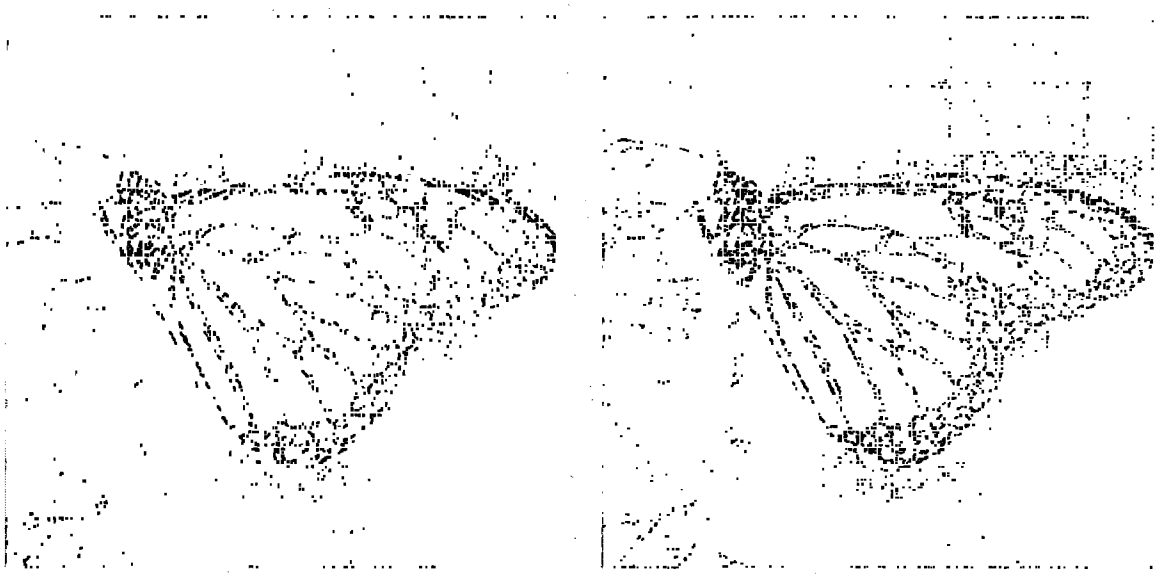


Figure 4.3: The selected blocksizes and the DA-DWT modes for the 512×512 image *Monarch*. Each macroblock contains 64×64 pixels. The Stage-1 directions $\mathbf{d}$ for the selected modes are shown on the left, and the Stage-2 directions $\check{\mathbf{d}}$ are shown on the right. In this example, the Lagrangian multiplier $\bar{\lambda}$ is determined by $Q_{\text{H.264}} = 34$, corresponding to a quantization stepsize of 32. The resulting overhead signaling these selections is coded at 0.005 bpp, and the rate to encode the coefficients with this quantization stepsize is 0.28 bpp with a reconstruction quality at 32.32 dB.

on heuristics and are hence in general suboptimal. The optimal update functions require rather intensive computation and therefore are not adopted in practice [84,85]. Consequently, although using smaller blocks for direction selection may provide better adaptation to the image content, the increased number of samples at block boundaries and the accompanied suboptimal update functions might on the contrary lead to a higher distortion. For this reason, in the DA-DWT we typically use macroblocks of 64×64 pixels, larger than the 16×16 macroblocks used in the DA-PBT as discussed in Sec. 3.2.1. An example of the selected blocksizes and modes are shown in Fig. 4.3.

4.2.2 Entropy Coding

Perhaps the most important motivation to use wavelet image coding is to obtain an embedded image representation, typically achieved by coding the transform coefficients with an embedded bitplane coder [166,171,183]. At the encoder, the coefficients

are coded bitplane by bitplane at high precision; the decoder can either retrieve the full bitstream to recover all the bitplanes encoded and hence a high-quality reconstruction, or only decode partial bitplanes, at a lower rate, to reconstruct at a reduced quality. This property is especially useful for progressive transmission of images over networks.

Since the DA-DWT coefficients can be arranged in the same way as the corresponding 2-D DWT coefficients, as they share the same number of subbands and the same number of coefficients in each subband, any embedded bitplane coder designed for the 2-D DWT may be used for the DA-DWT. In our implementation, we adopt the TCE coder for its open accessibility and superior performance [74, 173, 186].

To enable an embedded representation, the decoding rate, hence the decoding quality, is not known a priori at the encoder. To select the blocksizes and the DA-DWT modes, the choice of the Lagrangian multiplier $\bar{\lambda}$ as described in the previous section, however, must anticipate a certain decoding quality. On one hand, if the anticipated quality at the encoder is higher than the actual one, the resulting $\bar{\lambda}$ is smaller than the ideal value, leading to an unnecessarily detailed direction representation and an excessive signaling overhead. On the other hand, if the anticipated quality is lower, the consequent selections may not fulfill the full potential of the DA-DWT. As a compromise, we generally select $Q_{\text{H.264}} = 34$, corresponding to a quantization stepsize of 32, for the choice of $\bar{\lambda}$, as this usually leads to a reconstruction around 30 to 32 dB and below 0.4 bpp for typical images, where the visual improvements from the DA-DWT is most prominent, as we shall demonstrate in Sec. 4.4.

To encode the direction representation, for each block a 1-bit symbol is used to indicate whether Mode 0 is selected. If this is not the case, the selected mode is predicted from the blocks in the causal neighborhood, similar to the prediction mechanism described in Sec. 3.2.2, and the residual is coded using variable-length coding with a fixed codeword table. Additionally, if all the blocks in the causal neighborhood have selected Mode 0, the current selection is coded directly without prediction. For each macroblock, the selected blocksize is also coded with variable-length coding.

In the scope of our discussion, we consider only lossless coding of the selections.

The entirety of the signaling overhead is always transmitted to the decoder regardless of the decoding rate. At lower decoding rates, the overhead can take a significant portion of the total rate available, leaving limited rate for the transform coefficients. In [216], we have developed an approach to allow lossy coding of the selections in order to provide an embedded direction representation, analogous to the embedded representation of motion vectors in video coding [169, 212]. With this approach, the decoder is able to retrieve a coarser direction presentation, at a lower signaling rate, if the total decoding rate budget is limited.

4.2.3 Computational Complexity

To implement the DA-DWT, a straightforward method is to first carry out each of the 9 modes for the whole image and record the sum of absolute values of the resulting wavelet coefficients in every $\frac{S_B}{4} \times \frac{S_B}{4}$ block, the smallest blocksize adopted in the framework. Once the sum of absolute values are available, the direction selection process can be performed entirely based on these values. Last, the actual transform is applied using the selected blocksizes and modes to generate the final coefficients. As a result, 10 iterations of the wavelet transform are performed at the encoder. Assuming that, in a conventional wavelet-based image encoder, the transform contributes to around 50% of the total computation as reported in [1] for JPEG2000, an adaptive encoder based on the DA-DWT thus requires 5 to 6 times the computation as a conventional encoder. Although not included in our implementation, the complexity may be reduced by the approach described in Sec. 3.2.3 proposed for the DA-PBT. The approach first performs simple image analysis locally to identify the local directionality so that at most one directional mode, in addition to the conventional transform, is applied to every block. This can cut down the encoder complexity to less than twice the computation as a conventional encoder. Note that the decoder complexity is about the same as a conventional decoder since only the mode selected at the encoder is performed at the decoder.

4.3 Theoretical Analysis of DA-DWT

In this section, we provide theoretical analysis of the performance gain resulting from adapting the wavelet filtering directions. We first relate the frequency responses of the directional filters in the lifting steps with the underlying wavelet kernel. Using these frequency responses, the power spectral density (PSD) of the high-pass and the low-pass subbands is then related to the PSD of the image samples. The transform coding gain is then analyzed to quantify the performance gain against the conventional 2-D DWT.

4.3.1 Directional Filtering in Lifting

As described in Sec. 4.1, to apply the 2-D DWT to image samples $s[\mathbf{l}]$, $\mathbf{l} = (l_x, l_y)^T \in \mathbb{Z}^2$, using lifting, $s[\mathbf{l}]$ is first vertically subsampled into the even rows, $s_0[\mathbf{l}] = s[l_x, 2l_y]$, and the odd rows, $s_1[\mathbf{l}] = s[l_x, 2l_y+1]$. Similar to (4.1), the prediction and the update step in lifting can be expressed as

$$w_1[\mathbf{l}] = g_H \cdot (s_1[\mathbf{l}] - h_P^{\mathbf{d}}[\mathbf{l}] * s_0[\mathbf{l}]), \tag{4.8a}$$

$$w_0[\mathbf{l}] = g_L \cdot (s_0[\mathbf{l}] + g_H^{-1} \cdot (h_U^{\mathbf{d}}[\mathbf{l}] * w_1[\mathbf{l}])), \tag{4.8b}$$

where $w_0[\mathbf{l}]$ and $w_1[\mathbf{l}]$ denote the high-pass and the low-pass coefficients respectively, $h_P^{\mathbf{d}}[\mathbf{l}]$ and $h_U^{\mathbf{d}}[\mathbf{l}]$ denote the 2-D impulse response of the directional filters in the lifting steps along direction $\mathbf{d} = (d_x, d_y)^T$, the symbol $*$ denotes the 2-D convolution operation, and g_L and g_H are scaling factors. In the following discussion, functions denoted by an upper-case letter in the form of $A(e^{j\mathbf{\Omega}}) = A(e^{j\Omega_x}, e^{j\Omega_y})$ represent the 2-D discrete-space Fourier transform of the corresponding discrete-space signal $a[\mathbf{l}] = a[l_x, l_y]$. We start the analysis by deriving $H_P^{\mathbf{d}}(e^{j\mathbf{\Omega}})$ and $H_U^{\mathbf{d}}(e^{j\mathbf{\Omega}})$ as functions of the filtering direction $\mathbf{d}$ and the adopted wavelet kernel.

To facilitate the analysis, we define $\hat{s_0}[\mathbf{l}]$ as the upsampled $s_0[\mathbf{l}]$, i.e.,

$$\hat{s_0}[\mathbf{l}] = \hat{s_0}[l_x, l_y] = \begin{cases} s_0[l_x, l_y/2], & l_y \text{ is even,} \\ 0, & l_y \text{ is odd.} \end{cases} \tag{4.9}$$

Following the discussion in Sec. 4.1.2, we further define the 2-D directional prediction filters applied to $\hat{s}_0[\mathbf{l}]$ by the impulse response $\hat{h}_P^{\mathbf{d}}[\mathbf{l}]$,

$$\hat{h}_P^{\mathbf{d}}[\mathbf{l}] = \begin{cases} c_{P,k}, & \mathbf{l} = (2k+1)\mathbf{d}, \quad k = -K_P, \cdots, K_P - 1, \\ 0, & \text{otherwise}, \end{cases} \tag{4.10}$$

where K_P and $c_{P,k}$ are determined by the wavelet kernel adopted. In the conventional DWT, $\hat{h}_P^{\mathbf{d}_0}[\mathbf{l}]$, $\mathbf{d}_0 = (0,1)^T$, is always used, corresponding to vertical filtering with the prototype 1-D prediction filter, $\hat{h}_P[t]$, defined as

$$\hat{h}_P[t] = \hat{h}_P^{\mathbf{d}_0}[0,t], \quad t \in \mathbb{Z}. \tag{4.11}$$

Denote the 1-D discrete-space Fourier transform of $\hat{h}_P[t]$ by $\hat{H}_P(e^{j\Omega_t})$. We continue the analysis by relating $\hat{H}_P^{\mathbf{d}}(e^{j\boldsymbol{\Omega}})$ with $\hat{H}_P(e^{j\Omega_t})$. As defined in Sec. 4.1.2, d_x and d_y are coprime, i.e., their greatest common divisor is 1. From Bézout's identity [108], there is at least one direction, $\bar{\mathbf{d}} = (\bar{d}_x, \bar{d}_y)^T \in \mathbb{Z}^2$, such that

$$\bar{d}_x d_y - \bar{d}_y d_x = 1. \tag{4.12}$$

Therefore, a generator matrix defined as

$$\mathbf{W} = \begin{pmatrix} \bar{d}_x & d_x \\ \bar{d}_y & d_y \end{pmatrix} \tag{4.13}$$

is a one-to-one mapping from $\mathbb{Z}^2$ to $\mathbb{Z}^2$ [81], i.e.,

$$\{\, \mathbf{l}' \mid \mathbf{l}' = \mathbf{W}\mathbf{l}, \ \forall\, \mathbf{l} \in \mathbb{Z}^2 \} = \mathbb{Z}^2. \tag{4.14}$$

With $\mathbf{W}$, $\hat{h}_P^{\mathbf{d}}[\mathbf{l}]$ can be related to $\hat{h}_P^{\mathbf{d}_0}[\mathbf{l}]$ by

$$\hat{h}_P^{\mathbf{d}}[\mathbf{W}\mathbf{l}] = \hat{h}_P^{\mathbf{d}_0}[\mathbf{l}], \ \ \hat{h}_P^{\mathbf{d}}[\mathbf{l}] = \hat{h}_P^{\mathbf{d}_0}[\mathbf{W}^{-1}\mathbf{l}] \tag{4.15}$$

From (4.12), (4.13), and the fact that d_x and d_y are coprime, one can verify that (4.15) is consistent with the definition in (4.10).

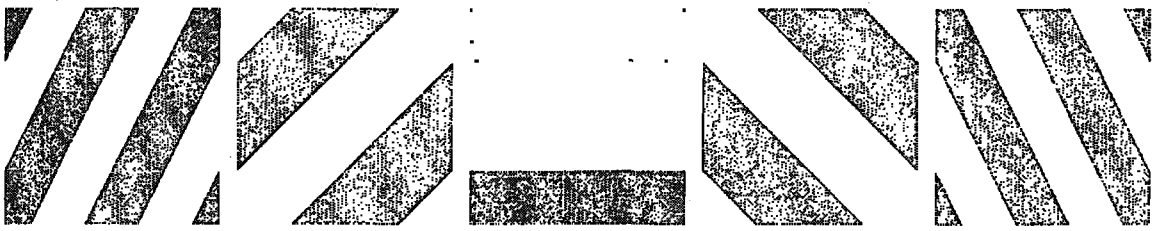


Figure 4.4: $\hat{H}_P^{\mathbf{d}}(e^{j\mathbf{\Omega}})$ in (4.18) using the sinc wavelet. Each plot represents the 2-D discrete-space frequency $-\pi \leq \Omega_x \leq \pi$ from left to right, and $-\pi \leq \Omega_y \leq \pi$ from bottom to top. Light gray and dark gray represent 1 and -1 respectively.

From (4.10) and (4.11), $\hat{h}_P^{\mathbf{d}_0}[\mathbf{l}] = \delta[l_x] \cdot \hat{h}_P[l_y]$, and it is straightforward that

$$\hat{H}_P^{\mathbf{d}_0}(e^{j\mathbf{\Omega}}) = \hat{H}_P(e^{j\Omega_y}). \tag{4.16}$$

Moreover, from (4.14), (4.15), and (4.16),

$$\begin{aligned} \hat{H}_P^{\mathbf{d}}(e^{j\mathbf{\Omega}}) &= \sum_{\mathbf{l}' \in \mathbb{Z}^2} \hat{h}_P^{\mathbf{d}}[\mathbf{l}'] e^{-j\mathbf{\Omega}^T \mathbf{l}'} = \sum_{\mathbf{l}' \in \mathbb{Z}^2} \hat{h}_P^{\mathbf{d}_0}[\mathbf{W}^{-1}\mathbf{l}'] e^{-j\mathbf{\Omega}^T \mathbf{l}'} \\ &= \sum_{\mathbf{l} = \mathbf{W}^{-1}\mathbf{l}' \in \mathbb{Z}^2} \hat{h}_P^{\mathbf{d}_0}[\mathbf{l}] e^{-j\mathbf{\Omega}^T \mathbf{W}\mathbf{l}} = \hat{H}_P^{\mathbf{d}_0}(e^{j\mathbf{W}^T\mathbf{\Omega}}) = \hat{H}_P(e^{j\mathbf{d}^T\mathbf{\Omega}}). \end{aligned} \tag{4.17}$$

As an example, using the sinc wavelet introduced in Sec. 4.1.3,

$$\hat{H}_P^{\mathbf{d}}(e^{j\mathbf{\Omega}}) = \begin{cases} 1, & q^{\mathbf{d}}(\Omega_x, \Omega_y) \text{ is even}, \\ -1, & q^{\mathbf{d}}(\Omega_x, \Omega_y) \text{ is odd}, \end{cases} \tag{4.18}$$

$$\hat{H}_U^{\mathbf{d}}(e^{j\mathbf{\Omega}}) = \tfrac{1}{2}\hat{H}_P^{\mathbf{d}}(e^{j\mathbf{\Omega}}), \tag{4.19}$$

where $q^{\mathbf{d}}(\Omega_x, \Omega_y) = \lfloor (d_x\Omega_x + d_y\Omega_y + \frac{\pi}{2})/\pi \rfloor$ and $\lfloor\ \rfloor$ denotes the floor function. Plots of $\hat{H}_P^{\mathbf{d}}(e^{j\mathbf{\Omega}})$ for the Stage-1 directions included in the 9 modes of the DA-DWT are shown in Fig. 4.4.

In the following derivation, we establish the relationship between the frequency response of the filter in (4.8a), $H_P^{\mathbf{d}}(e^{j\mathbf{\Omega}})$, with the equivalent filter in the upsampled domain in (4.10), $\hat{H}_P^{\mathbf{d}}(e^{j\mathbf{\Omega}})$. From (4.8) and (4.10), convolving $h_P^{\mathbf{d}}[\mathbf{l}]$ with $s_0[\mathbf{l}]$ is equivalent to convolving $\hat{h}_P^{\mathbf{d}}[\mathbf{l}]$ with $\hat{s}_0[\mathbf{l}]$ followed by a 1-pixel vertical advance and

vertical downsampling, i.e.,

$$(h_P^{\mathbf{d}} * s_0)[\mathbf{l}] = (\hat{h}_P^{\mathbf{d}} * \hat{s_0})[l_x, 2l_y + 1]. \tag{4.20}$$

Hence,

$$H_P^{\mathbf{d}}(e^{j\mathbf{\Omega}}) \cdot S_0(e^{j\mathbf{\Omega}}) = \frac{1}{2} \sum_{k=0}^{1} \hat{H}_P^{\mathbf{d}}(e^{j\Omega_x}, e^{j(\frac{\Omega_y}{2}+k\pi)}) \cdot e^{j(\frac{\Omega_y}{2}+k\pi)} \cdot \hat{S}_0(e^{j\Omega_x}, e^{j(\frac{\Omega_y}{2}+k\pi)}). \tag{4.21}$$

From (4.9),

$$\hat{S}_0(e^{j\mathbf{\Omega}}) = S_0(e^{j\Omega_x}, e^{j2\Omega_y}). \tag{4.22}$$

In addition, from the fact that d_y is always an odd integer and (4.10), $\hat{h}_P^{\mathbf{d}}[\mathbf{l}] = \hat{h}_P^{\mathbf{d}}[l_x, l_y] = 0$ if l_y is even. Since $e^{-jk\pi l_y} = (-1)^{l_y}$ if k is an odd integer,

$$\begin{aligned} \hat{H}_P^{\mathbf{d}}(e^{j\mathbf{\Omega}}) &= \sum_{l_y \in \mathbb{Z}} \sum_{l_x \in \mathbb{Z}} (\hat{h}_P^{\mathbf{d}}[l_x, l_y] e^{-j\Omega_x l_x} e^{-j\Omega_y l_y} \cdot -e^{-jk\pi l_y}) \\ &= -\hat{H}_P^{d}(e^{j\Omega_x}, e^{j(\Omega_y + k\pi)}), \quad \forall\, k : k \text{ is odd.} \end{aligned} \tag{4.23}$$

Combining (4.21), (4.22) and (4.23), we obtain

$$H_P^{\mathbf{d}}(e^{j\mathbf{\Omega}}) = \frac{1}{2} \sum_{k=0}^{1} (-1)^k e^{j\frac{\Omega_y}{2}} \hat{H}_P^{\mathbf{d}}(e^{j\Omega_x}, e^{j(\frac{\Omega_y}{2}+k\pi)}) = e^{j\frac{\Omega_y}{2}} \hat{H}_P^{\mathbf{d}}(e^{j\Omega_x}, e^{j\frac{\Omega_y}{2}}). \tag{4.24}$$

Finally, combining (4.17) and (4.24), we obtain

$$H_P^{\mathbf{d}}(e^{j\mathbf{\Omega}}) = e^{j\frac{\Omega_y}{2}} \hat{H}_P(e^{j(d_x\Omega_x + d_y\frac{\Omega_y}{2})}). \tag{4.25}$$

With a similar derivation, the frequency response of the filter in (4.8b), $H_U^{\mathbf{d}}(e^{j\mathbf{\Omega}})$, can be obtained as

$$H_U^{\mathbf{d}}(e^{j\mathbf{\Omega}}) = e^{-j\frac{\Omega_y}{2}} \hat{H}_U(e^{j(d_x\Omega_x + d_y\frac{\Omega_y}{2})}), \tag{4.26}$$

where $\hat{H}_U(e^{j\Omega_t})$ is the frequency response of the prototype 1-D update filter.

4.3.2 Coefficient Power Spectral Density

Assume $s[\mathbf{l}]$ is a stationary 2-D discrete-space random field. Denote the 2-D discrete-space autocorrelation function of $s[\mathbf{l}]$ by $R_{ss}[\Delta\mathbf{l}] = E\{s[\mathbf{l}+\Delta\mathbf{l}]s^*[\mathbf{l}]\}$ where $s^*[\mathbf{l}]$ denotes the complex conjugate of $s[\mathbf{l}]$, and $s[\mathbf{l}]$ is real so that $s^*[\mathbf{l}] = s[\mathbf{l}]$. The 2-D discrete-space PSD of $s[\mathbf{l}]$ is denoted by $\Phi_{ss}(e^{j\mathbf{\Omega}})$ (with the shorthand notation Φ_{ss}). Since $s_0[\mathbf{l}]$ and $s_1[\mathbf{l}]$ are vertical subsamples of $s[\mathbf{l}]$ with different phases,

$$R_{s_0s_0}[\Delta\mathbf{l}] = R_{s_1s_1}[\Delta\mathbf{l}] = R_{ss}[\Delta l_x, 2\Delta l_y], \tag{4.27}$$

$$R_{s_0s_1}[\Delta\mathbf{l}] = R_{ss}[\Delta l_x, 2\Delta l_y - 1]. \tag{4.28}$$

Therefore,

$$\Phi_{s_0s_0}(e^{j\mathbf{\Omega}}) = \Phi_{s_1s_1}(e^{j\mathbf{\Omega}}) = \tfrac{1}{2}\sum_{k=0}^{1}\Phi_{ss}(e^{j\Omega_x}, e^{j(\frac{\Omega_y}{2}+k\pi)}), \tag{4.29}$$

$$\Phi_{s_0s_1}(e^{j\mathbf{\Omega}}) = \tfrac{1}{2}\sum_{k=0}^{1}(-1)^k e^{-j\frac{\Omega_y}{2}}\Phi_{ss}(e^{j\Omega_x}, e^{j(\frac{\Omega_y}{2}+k\pi)}). \tag{4.30}$$

From (4.8), (4.25), (4.26), and the property that $\Phi_{ss}(e^{j\mathbf{\Omega}})$ is real, the PSD of the high-pass and the low-pass subbands when wavelet filtering is applied along direction $\mathbf{d}$, denoted by $\Phi^{\mathbf{d}}_{w_1w_1}$ and $\Phi^{\mathbf{d}}_{w_0w_0}$ respectively, can be expressed as

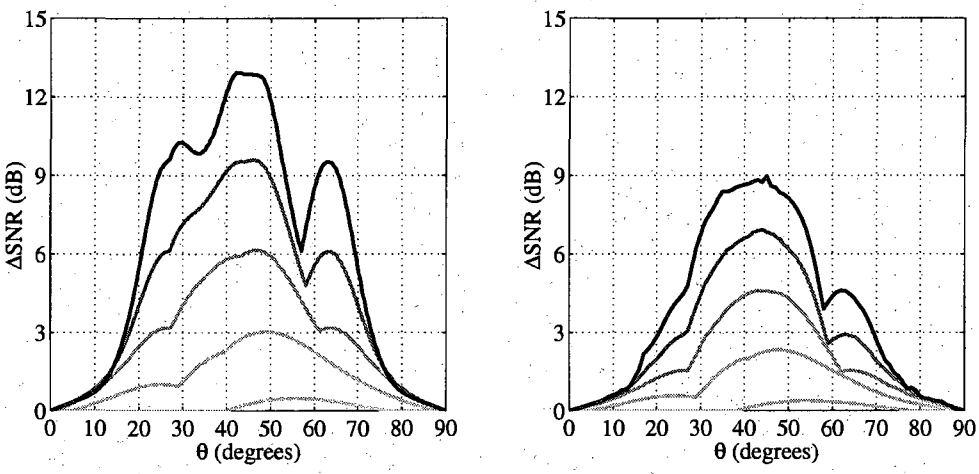
$$\Phi^{\mathbf{d}}_{w_1w_1} = g_H^2 \cdot ((1+|H_P^{\mathbf{d}}|^2)\Phi_{s_0s_0} - 2H_P^{\mathbf{d}}\Phi_{s_0s_1}), \tag{4.31}$$

$$\Phi^{\mathbf{d}}_{w_0w_0} = g_L^2 \cdot ((|1-H_U^{\mathbf{d}}H_P^{\mathbf{d}}|^2 + |H_U^{\mathbf{d}}|^2)\Phi_{s_0s_0} + 2(1 - H_U^{\mathbf{d}}H_P^{\mathbf{d}})H_U^{\mathbf{d}*}\Phi_{s_0s_1}). \tag{4.32}$$

Note that $H_P^d\Phi_{s_0s_1}$, $H_U^dH_P^d$, and $H_U^{d*}\Phi_{s_0s_1}$ are all real.

4.3.3 Transform Coding Gain

The derivation in Sec. 4.3.1 and Sec. 4.3.2 relates the PSD of the two subbands resulting from Stage 1 of the DA-DWT to the PSD of the source image pixels. It is



(a) 5/3 filter pair (b) sinc wavelet

Figure 4.5: Transform coding gain improvement from 3 levels of the DA-DWT upon the conventional 2-D DWT using (a) the 5/3 filter pair, and (b) the sinc wavelet. The source PSD is generated assuming elliptic random fields with $\lambda_a = -\ln(0.8) f_s$, $\lambda_b = \lambda_a / k_b$ where $k_b = 2, 4, 8, 16, 32$, and θ from 0° to 90°. A darker line corresponds to a larger k_b, i.e., stronger directionality.

straightforward to extend the derivation to Stage 2 of the DA-DWT as well as multiple levels of the transform so that the PSD of any subband can be derived recursively from the source PSD, given the selected mode in each level and the underlying wavelet kernel. The variance in the subband can then be computed by averaging the corresponding PSD over $-\pi \leq \Omega_x < \pi$ and $-\pi \leq \Omega_y < \pi$ numerically, and the calculation of the transform coding gain defined in Sec. 3.4.1 follows.

The transform coding gain improvement from the DA-DWT upon the conventional 2-D DWT in decibel (dB), i.e., $10 \cdot \log_{10} \frac{G_{DA-DWT}}{G_{DWT}}$, is plotted in Fig. 4.5 assuming the source is the Gaussian elliptic random field described in Sec. 2.3.1, using both the 5/3 filter pair and the sinc wavelet. Note that although the formulation in (2.21) is only applicable to orthonormal transforms, it can be extended to biorthogonal transforms, such as the 5/3 filter pair, by considering the energy expansion factors in wavelet synthesis [185]. Nevertheless, this additional consideration can be excluded in our calculation since we are only interested in the coding gain ratio between two transforms adopting the same wavelet kernel, hence sharing the same energy expansion factors. We assume three levels of the transforms are applied to the source, and

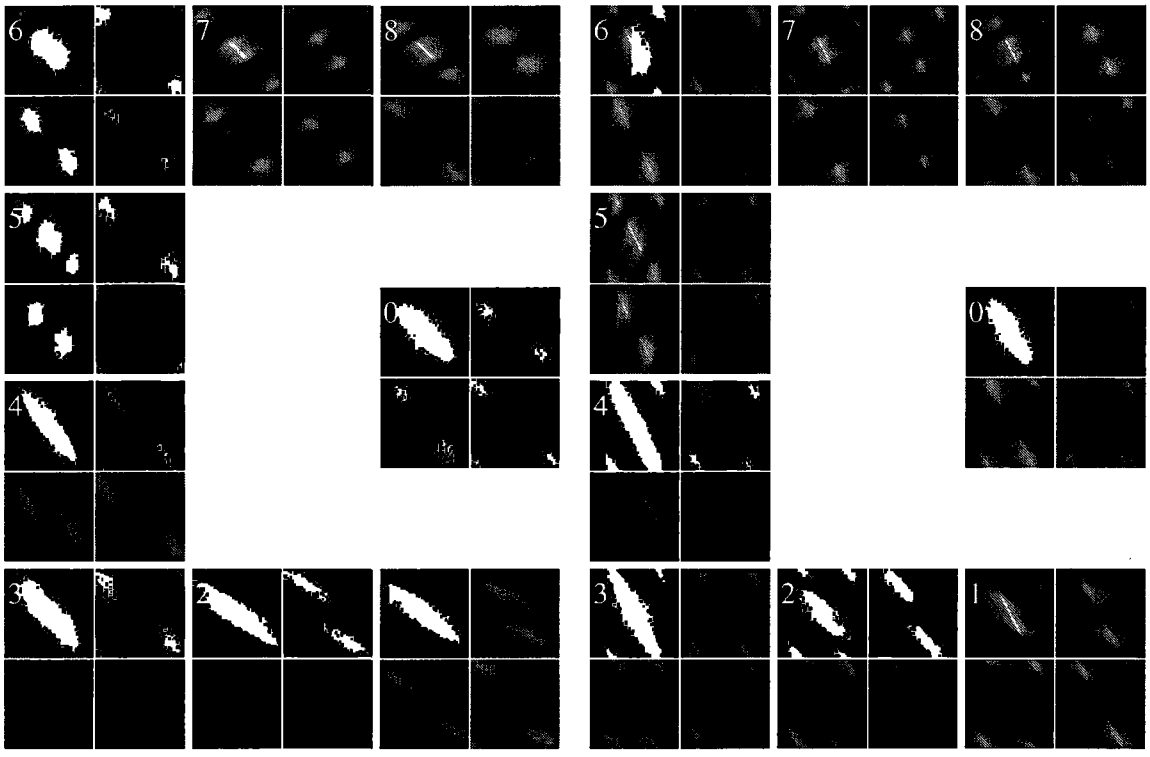
the DA-DWT always selects the mode with the highest transform coding gain. In Fig. 4.5, we consider only the elliptic random fields oriented from 0° to 90° due to the symmetry between Mode 1–4 and Mode 5–8 of the DA-DWT (Fig. 4.1-(a)).

In Fig. 4.5, as expected, the DA-DWT leads to a larger improvement for sources with stronger directionality. The peaks around 30° and 45° in Fig. 4.5-(a) come from Mode 2 and Mode 3 respectively, and the peak at 63° and its symmetric counterpart around 27° are contributed by Mode 4 and Mode 1 respectively. For further illustration, the PSD of two examples of the elliptic random fields are shown in Fig. 4.6. In Fig. 4.7, these PSDs are filtered by the 2-D analysis filters of one level of the DA-DWT using the 5/3 filter pair, whose frequency responses are shown in Fig. 4.2-(a). The intensity in the plots of the filtered PSD directly indicates the energy contained in each subband. For the example in Fig. 4.6-(a) and Fig. 4.7-(a), most energy is captured in the LL subband in Mode 3. Therefore, Mode 3 is selected as it gives the best energy concentration and hence the highest transform coding gain. Similarly, Mode 4 is selected in Fig. 4.7-(b) for the example in Fig. 4.6-(b). Note that the aliasing components in Fig. 4.6-(b) due to undersampling of sharp image textures are aligned with the passband of the LL subband in Mode 4 as shown in Fig. 4.2. Hence, compared to other modes, in Mode 4 even the energy in the aliasing components is concentrated in the LL subband, resulting in a higher transform coding gain.

Comparing Fig. 4.5-(a) and Fig. 4.5-(b), a larger improvement is observed when the 5/3 filter pair is adopted due to the overlapping subbands. In the conventional transform (Mode 0) using the 5/3 filter pair, the source energy residing in the overlapping regions is leaked into multiple subbands, hampering energy concentration. With a properly selected directional mode, most energy is contained in the interior of one subband, further away from the overlapping regions. This significantly reduces the energy leakage problem and thus further improves energy concentration. Using the sinc wavelet, the subbands do not overlap and therefore the directional modes do not benefit from this improvement.

(a) $\lambda_a = -\ln(0.8)f_s$, $\lambda_b = \lambda_a/8$, $\theta = 45°$ (b) $\lambda_a = -\ln(0.8)f_s$, $\lambda_b = \lambda_a/32$, $\theta = 63°$

Figure 4.6: The PSD of two examples of the elliptic random field in logarithmic scale.



(a) Mode 3 selected (b) Mode 4 selected

Figure 4.7: The PSD examples in Fig. 4.6 filtered by the 2-D analysis filters of the 9 DA-DWT modes using the 5/3 filter pair. Mode 3 and Mode 4 are selected for (a) and (b) respectively as they lead to the highest transform coding gain.

4.3.4 Rate-Distortion Performance

Extending the analysis in Sec. 2.3.2 that models the image-wise rate-distortion performance of the 8×8 KLT and the 8×8 DCT using the proposed statistical image model, in this section we derive the model performance of the DA-DWT. We assume

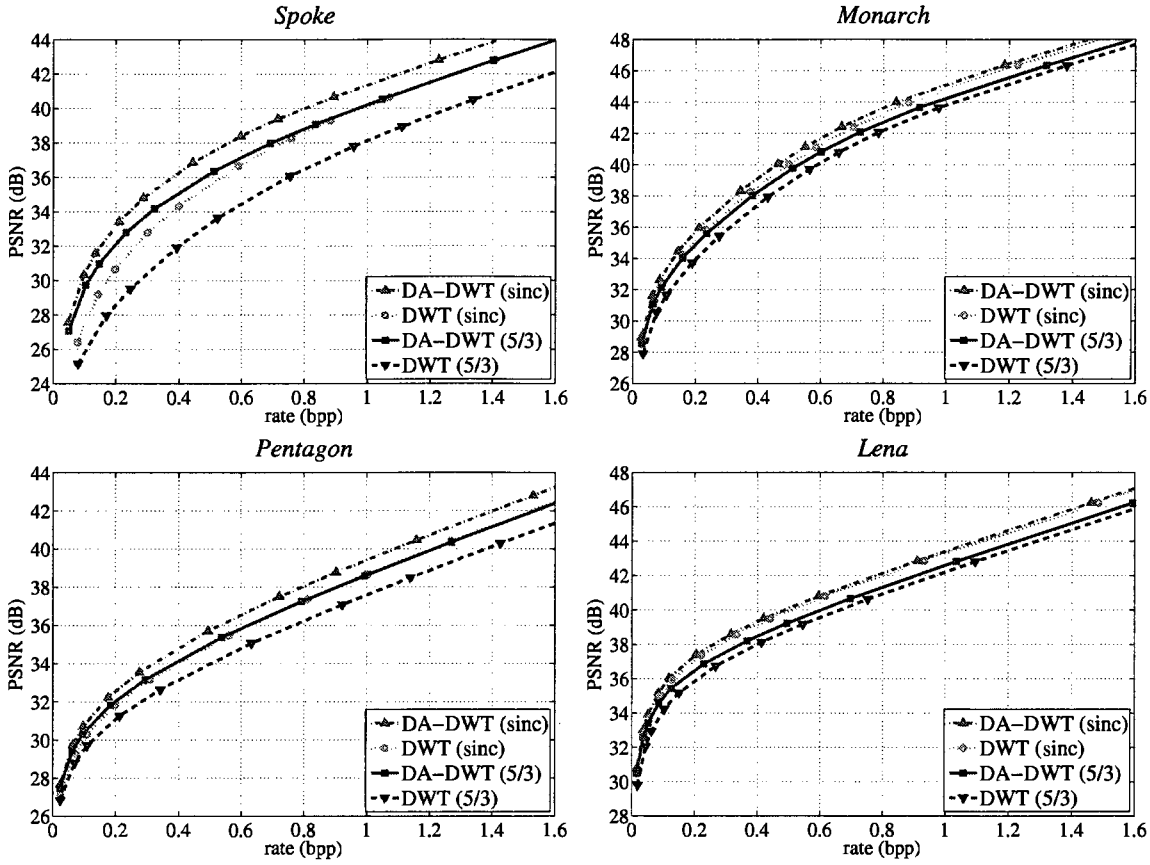


Figure 4.8: Model rate-distortion performance of the DA-DWT and the conventional 2-D DWT using the sinc wavelet and the 5/3 filter pair.

three levels of the DA-DWT are applied to the image, followed by an additional level of the conventional 2-D DWT to further decompose the resulting low-pass subband, same as in the actual experiments to be discussed in Sec. 4.4.

As explained in Sec. 2.3.2 and in the beginning of Sec. 4.3.3, for every mode of the DA-PWT, the model subband variances around a block can be derived from the image model. Locally around a block, we assume that every subband is coded independently from others, and the coding performance is ideal in the sense that it achieves the rate-distortion function of Gaussian memoryless sources. As a result, the local rate-distortion performance around the b-th block in the image using mode

m can be expressed as

$$R_b^{(m)}(\lambda) = \sum_{s=0}^{S-1} \eta_s \max\{0, \frac{1}{2}\log_2 \frac{\sigma_{b,s}^{2,(m)}}{\lambda}\}, \tag{4.33}$$
$$D_b^{(m)}(\lambda) = \sum_{s=0}^{S-1} \eta_s \min\{\lambda, \sigma_{b,s}^{2,(m)}\},$$

where S denotes the number of subbands and $S = 13$ for the 4-level transform, $\sigma_{b,s}^{2,(m)}$ denotes the variance of subband s around the b-th block using mode m, η_b denotes the ratio between the number of coefficients in subband b and the total number of coefficients, and λ controls the rate-distortion trade-off. For this block, the mode of the DA-DWT is selected by minimizing $D_b^{(m)}(\lambda) + 2\ln 2 \cdot \lambda R_b^{(m)}(\lambda)$ as for the DA-PBT explained in Sec. 3.4.3. Denote the mode selected for the b-th block by m_b and the number of blocks in the image by L_B, the image-wise performance is then

$$R(\lambda) = \frac{1}{L_B}\sum_{b=0}^{L_B-1} R_b^{(m_b)}(\lambda), \quad D(\lambda) = \frac{1}{L_B}\sum_{b=0}^{L_B-1} D_b^{(m_b)}(\lambda), \tag{4.34}$$

plotted in Fig. 4.8 for the images shown in Fig. 2.1 using both the 5/3 filter pair and the sinc wavelet. The model performance of 4 levels of the conventional 2-D DWT, which can also be expressed by (4.34) with $m_b = 0$ for every block, is included for reference.

For the 5/3 filter pair, at the same rate the DA-DWT outperforms the 2-D DWT by up to around 4 dB for *Spoke*, 1 dB for *Monarch*, 1.2 dB for *Pentagon* and 0.7 dB for *Lena*. Clearly, the DA-DWT delivers more improvements over the 2-D DWT for images containing more directional textures. The model performance gain closely follows the actual gain obtained by experiments, and this will be discussed in Sec. 4.4. In Fig. 4.8, for the sinc wavelet that requires an infinite support, the improvement by the DA-DWT over the 2-D DWT is less than that for the 5/3 filter pair, consistent with the transform coding gain analysis in Sec. 4.3.3. For *Spoke* and *Pentagon*, the performance of the DA-DWT using the 5/3 filter pair matches that of the 2-D DWT using the sinc wavelet. This suggests that instead of using a complicated kernel

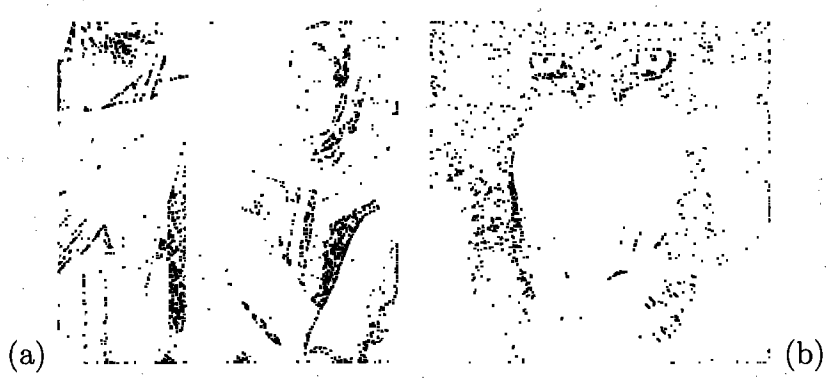


Figure 4.9: 512×512 8-bit grayscale image (a) *Barbara* and (b) *Mandrill*

that may require more computation, varying the filtering direction of a simple kernel according to the image content can be a better alternative.

4.4 Experimental Results

In the experimental results reported in this section, in addition to the test images shown in Fig. 2.1, we further include the two images in Fig. 4.9. Three levels of the DA-DWT are first applied to the image, followed by one additional level of the conventional 2-D DWT to further decompose the resulting low-pass subband. The three levels of the DA-DWT share the same set of direction selections, and the macroblock-size is set to 64×64 pixels as discussed in Sec. 4.2.1. The 5/3 filter pair is adopted for its simplicity and also for providing a direct comparison with the model performance in Fig. 4.8. The DA-DWT can also be applied with other wavelet kernels such as the $(6, 6)$ interpolating wavelet and the 9/7 filter pair as described in [26, 53, 129].

The compression performance of the DA-DWT for the test images is shown in Fig. 4.10. In addition, the performance of four levels of the conventional 2-D DWT, equivalent to always selecting Mode 0 in the DA-DWT, is also shown for comparison. For both transforms, the coefficients are encoded by the TCE embedded bitplane coder proposed in [186] as described in Sec. 4.2.2. The wavelet transforms, together with bitplane coding, provide an embedded image representation such that different decoding qualities are obtained by truncating the single bitstream from the encoder at different rates. To choose the Lagrangian multiplier $\bar{\lambda}$ for direction selection,

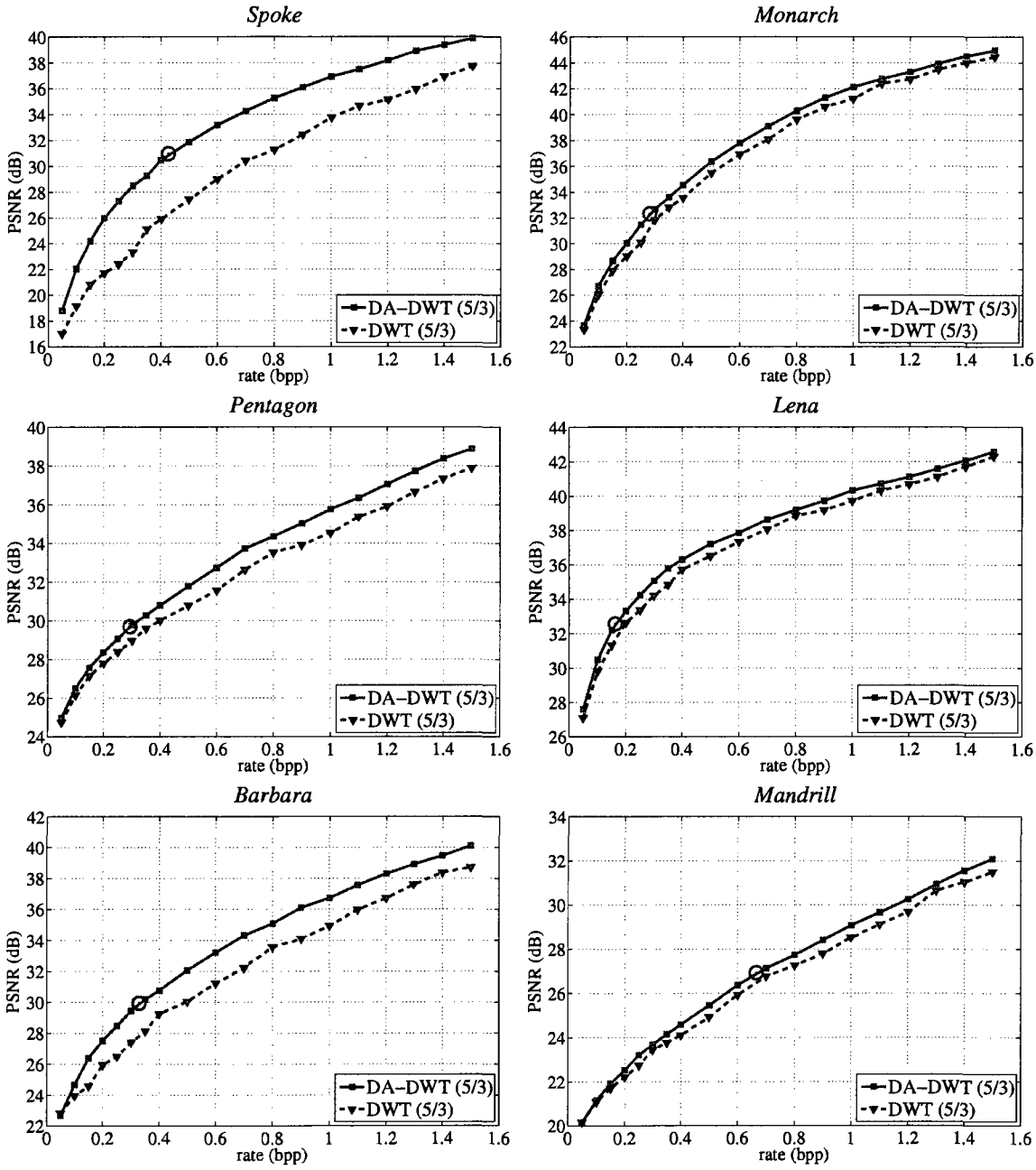


Figure 4.10: Rate-distortion performance of the DA-DWT and the conventional 2-D DWT using the 5/3 filter pair.

we always use $Q_{\text{H.264}} = 34$, equivalent to a target quantization stepsize of 32, as explained in Sec. 4.2.2. The decoding rate and distortion obtained by decoding the embedded bitstream up to the bitplane corresponding to this quantization stepsize is indicated by the circular marker on each DA-DWT curve in Fig. 4.10, usually taking place around 30 to 32 dB and below 0.4 bpp where the improvement in visual quality using the DA-DWT is most prominent as we shall demonstrate in Fig. 4.11, Fig. 4.12 and Fig. 4.13. The resulting signaling overhead ranges from 0.005 bpp (*Monarch*) to 0.01 bpp (*Mandrill*), and this rate is included in the rate calculation in Fig. 4.10.

Comparing the model performance in Fig. 4.8 with the actual performance in Fig. 4.10, at the same rate the model reconstruction quality is typically higher than the actual one by 2 to 4 dB. This is mainly due to the assumptions in the analysis in Sec. 4.3.4 that local statistics of the subband coefficients are known and the performance of coefficient coding is ideal. Although the practical TCE bitplane coder we adopted exploits the correlation across subbands to some extent [186], an advantage over the independent coding assumption also included in the analysis, its performance can still be far from the analytical coding scheme. In particular, the performance difference is most significant for *Spoke*, with an average around 4 dB, due to the highly varying local statistics between the foreground spoke and the background that are difficult to estimate for the adopted practical coder. Nevertheless, the main purpose of the analysis is to quantify the performance gain from the DA-DWT over the 2-D DWT in a general transform coding framework, and the actual gain, i.e., the gap between the two curves in Fig. 4.10, is indeed closely captured by the analysis.

As discussed in Sec. 1.2, the non-adaptive directional transforms in the literature such as the steerable pyramid [75, 174], the complex wavelet transform [70, 114], the ridgelet transform [17, 61], the curvelet transform [65, 178] and the contourlet transform [62] have shown certain improvement over the non-directional 2-D DWT in terms of nonlinear approximation performance. However, none has reported rate-distortion coding improvement for typical test images. Unlike the non-adaptive directional transforms, from Fig. 4.10, the DA-DWT outperforms the 2-D DWT by up to 5.1 dB at the same rate for *Spoke*, 1.4 dB for *Monarch*, 1.2 dB for *Pentagon*, 1.0 dB for *Lena*, 2.1 dB for *Barbara* and 0.6 dB for *Mandrill*. In addition to the natural images in the

test set, significant gain of more than 2 dB is also observed for fingerprint images, as well as medical images where sharp edges occur at the boundary between the object of interest and the background. In general, the performance gain is larger for images rich of features with large intensity transition in one direction and small variation in the orthogonal direction, again consistent with the transform coding gain analysis presented in Sec. 4.3.3.

For each test image, two 128×128 regions reconstructed at 0.05, 0.1, 0.2, 0.3, 0.4 and 0.5 bpp are included in Fig. 4.11, Fig. 4.12 and Fig. 4.13 to demonstrate the difference in visual quality between the DA-DWT and the 2-D DWT. In general, at lower rates the reconstruction from the DA-DWT exhibits brushstroke-like artifacts along image features. Different from the typical checkerboard artifacts resulting from the conventional transform, for instance as in the reconstructed *Spoke* and *Monarch* in Fig. 4.11, the brushstroke artifacts follow the geometric flow in the image, best observed at the stripes in *Barbara* and the hair in *Mandrill* in Fig. 4.13, and therefore better preserve the geometric structure. Hence, at low rates the DA-DWT can readily delineate the image content, making it a superior candidate for progressive transmission of images, where a low-quality preview is first reconstructed and, as more data are received, refined to achieve higher qualities.

4.5 Summary

We have proposed a new direction-adaptive discrete wavelet transform (DA-DWT) for image coding. The DA-DWT provides an efficient representation for directional image features such as edges and lines. Using a lifting structure, the DA-DWT is able to locally adapt the transform directions to the image content while ensuring the reversibility of the transform. In our experiments, the DA-DWT outperforms the conventional 2-D DWT by around 1 to 2 dB in PSNR for typical test images, and by up to 5.1 dB for special classes of images, consistent with the performance improvement derived from theoretical analysis using an image model.

Although the transform directions are selected on a block-by-block basis, filtering

extends across block boundaries so that blocking artifacts are absent in the reconstruction. At low rates, the reconstruction error of the DA-DWT exhibits brushstroke-like artifacts. Different from the ringing and checkerboard artifacts typically observed in the conventional transform, the brushstroke artifacts better preserve the geometric structure in the image. The ability to better depict the image content at low rates, together with embedded coding of the coefficients, makes the DA-DWT especially suitable for progressive transmission of images.

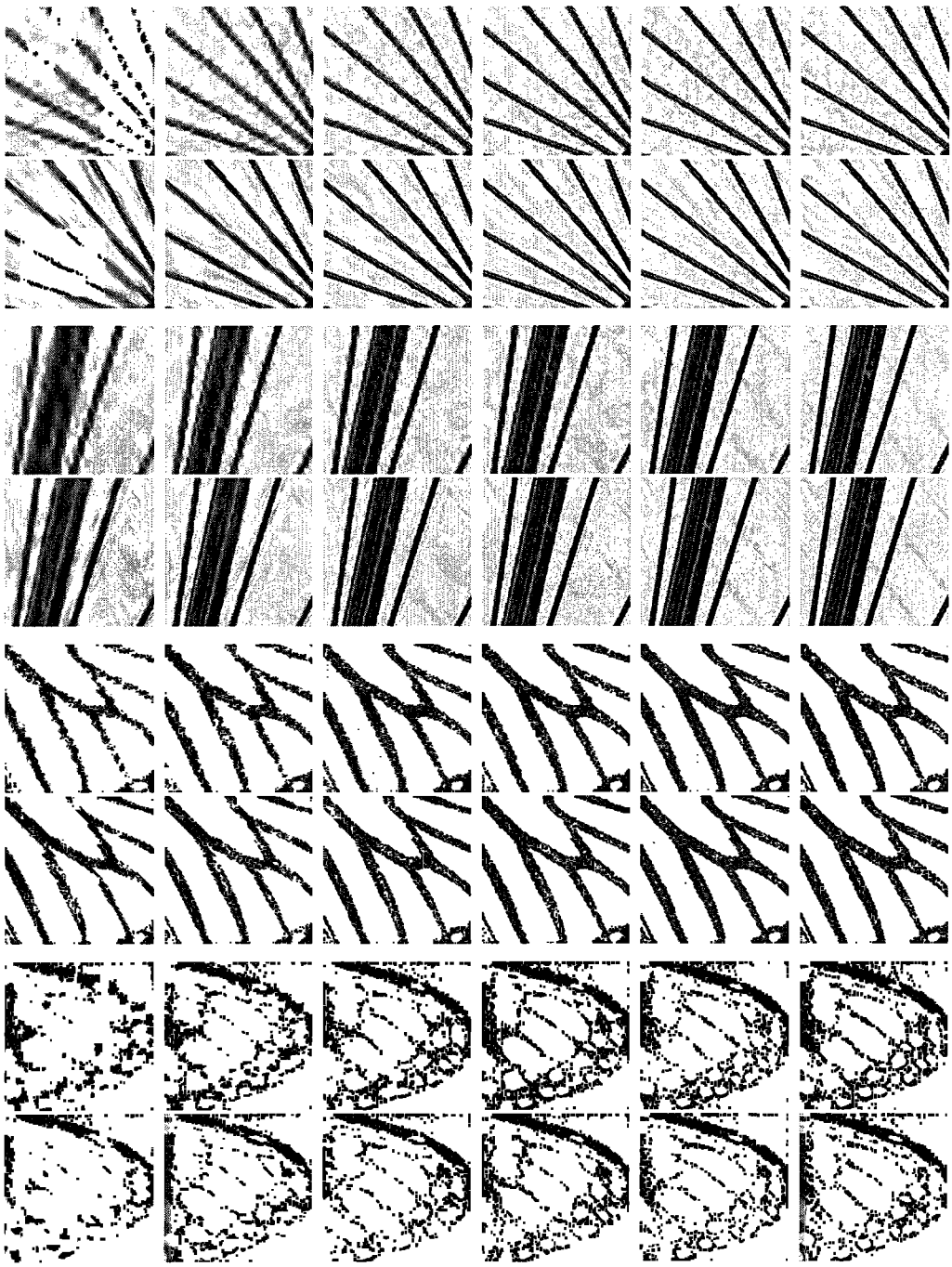


Figure 4.11: Each group of 12 plots corresponds to a 128×128 region in *Spoke* and *Monarch*. In each group, the reconstructions from the conventional 2-D DWT and the DA-DWT using the 5/3 filter pair are shown at the top and the bottom row respectively, decoded at 0.05, 0.1, 0.2, 0.3, 0.4 and 0.5 bpp from left to right.

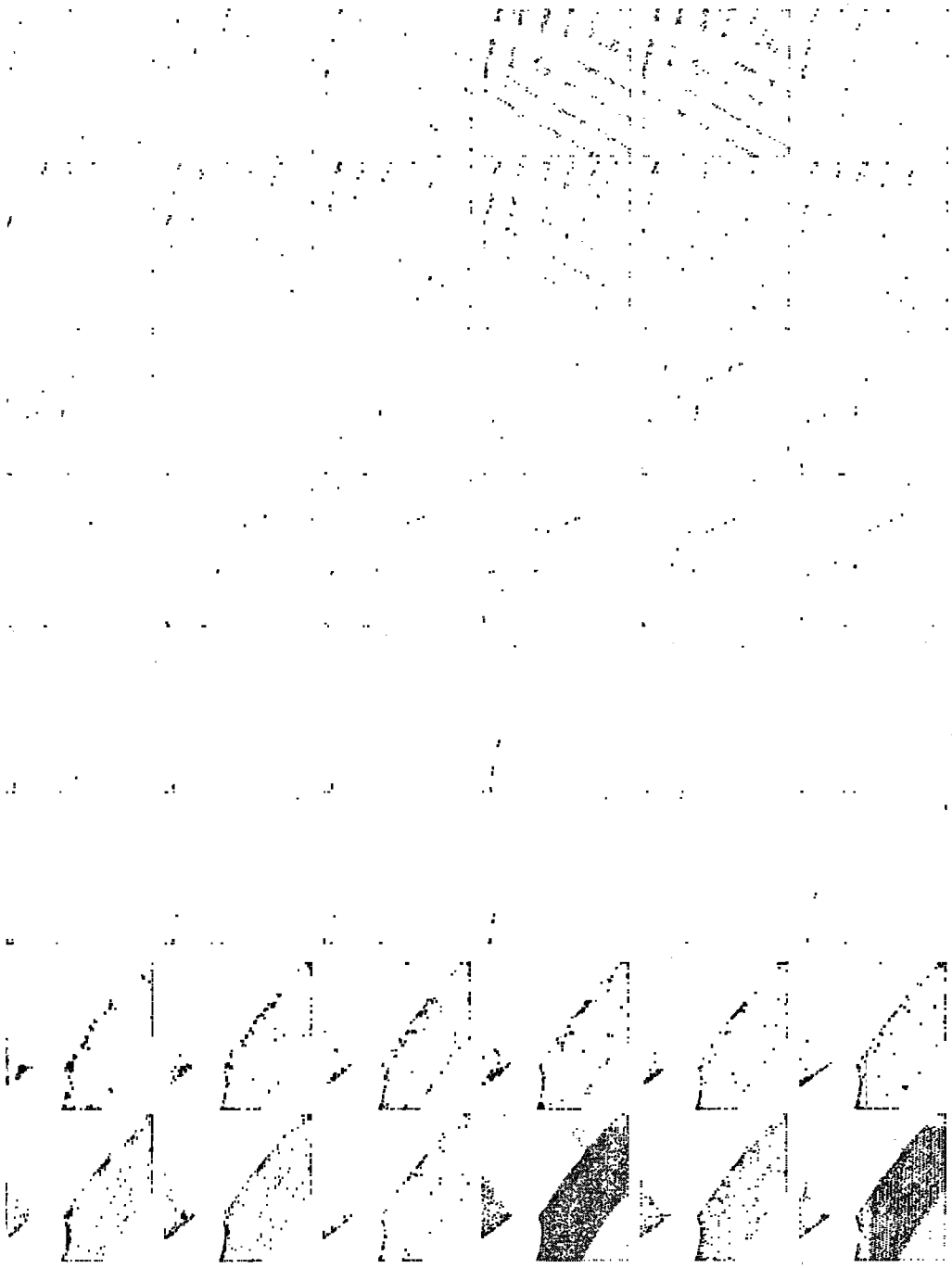


Figure 4.12: Each group of 12 plots corresponds to a 128 × 128 region in *Pentagon* and *Lena*. In each group, the reconstructions from the conventional 2-D DWT and the DA-DWT using the 5/3 filter pair are shown at the top and the bottom row respectively, decoded at 0.05, 0.1, 0.2, 0.3, 0.4 and 0.5 bpp from left to right.

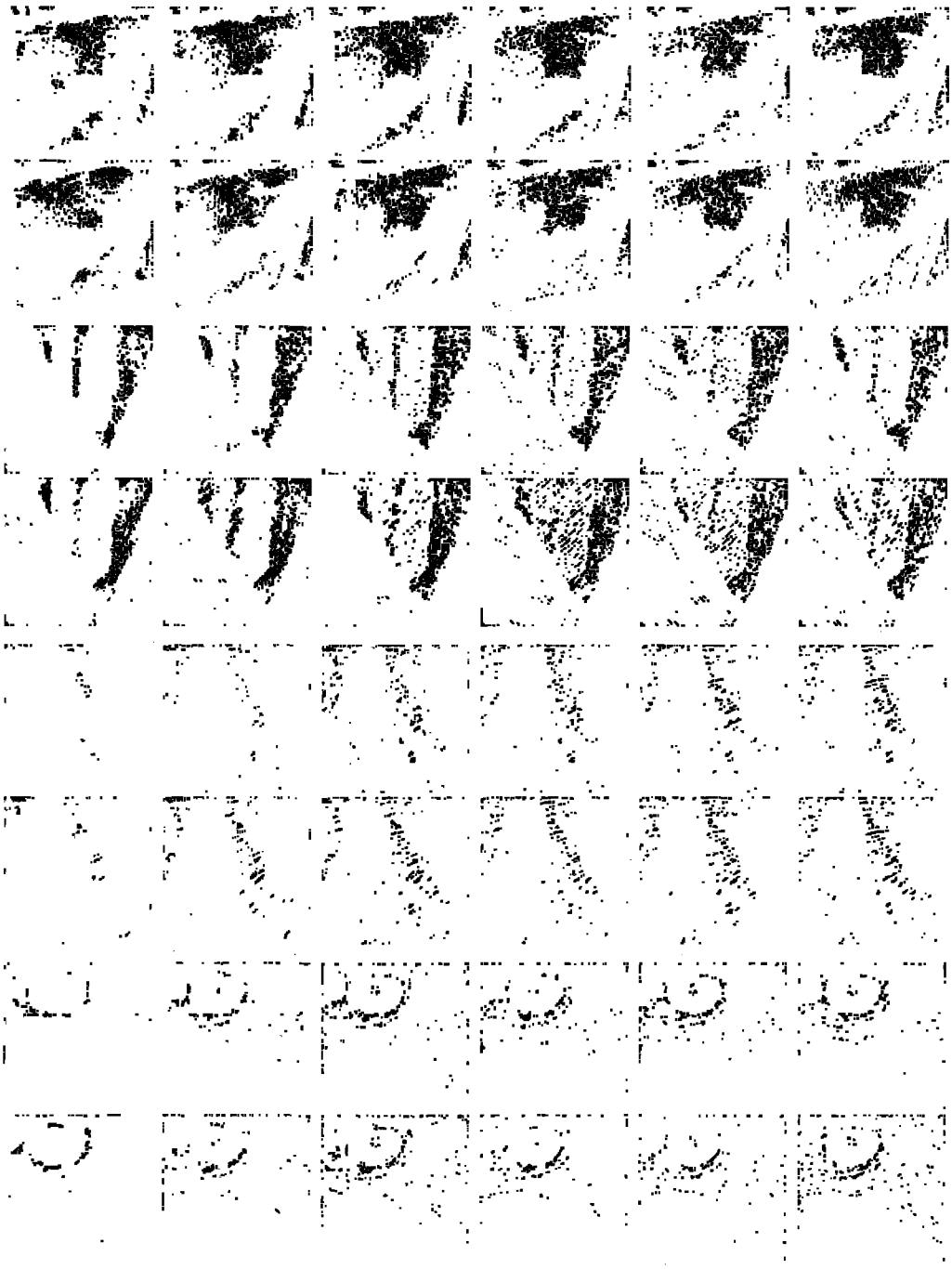


Figure 4.13: Each group of 12 plots corresponds to a 128×128 region in *Barbara* and *Mandrill*. In each group, the reconstructions from the conventional 2-D DWT and the DA-DWT using the 5/3 filter pair are shown at the top and the bottom row respectively, decoded at 0.05, 0.1, 0.2, 0.3, 0.4 and 0.5 bpp from left to right.

Chapter 5

Conclusions

In this dissertation, two new direction-adaptive transforms are proposed to improve the compression efficiency of practical image coding systems: the direction-adaptive partitioned block transform (DA-PBT) and the direction-adaptive discrete wavelet transform (DA-DWT), together with a novel image model that facilitates theoretical analysis of the coding performance. The direction-adaptive transforms adapt the selection of basis functions according to local directionality in images to efficiently represent directional image features such as edges and lines. Both new transforms are shown to be superior candidates for image coding to their non-adaptive counterparts.

The image model represents an image by a mixture of texture sources to cope with locally varying statistics. Each texture source is a 2-D stationary random field composed of an irregular and a regular texture component, characterized by the elliptic random field and the periodic-wave random field respectively. Using the texture sources, any image neighborhood centered in a particular image block is modeled as a segment of a realization of the texture source associated with the block, mixed with additive white noise and normalized by the local variance and the local mean around the block. For a given image, an iterative clustering algorithm is proposed to optimize the parameters of the texture sources, the association between the image blocks and the texture sources, and the white-noise level in each block. Using the model, theoretical analysis shows that using an adaptive transform consisting of the

KLTs derived from the texture sources instead of a fixed transform offers more improvement in terms of transform coding gain for textures with stronger directionality. The analysis also indicates that a substantial gain in rate-distortion performance can be expected around sharp directional features, rendered by the directional bases of the adaptive transform aligned with the image features.

To provide directional bases for a practical transform, the DA-PBT is proposed for coding of images and video sequences. The DA-PBT outperforms the conventional 2-D DCT by more than 2 dB in PSNR for images with pronounced directional features. Since it avoids the typical ringing and checkerboard artifacts of the 2-D DCT, subjective improvements are even larger than indicated by the rate-distortion performance. The DA-PBT also outperforms a previously proposed directional DCT while demanding less computation, owing to an improved directional selectivity, direction-adaptive partitioning and better coefficient ordering for entropy coding. To exploit the frequency response of the human visual system, quantization matrices can be used with the DA-PBT. We show a straightforward way to transform JPEG quantization matrices into the directional transform space, thus avoiding cumbersome subjective tests. The DA-PBT can also be combined with predictive coding, either directional prediction within the image, where the direction selection for the prediction and the transform can be elegantly combined, or interframe prediction for video. It is not surprising that the gains of both techniques are not additive, as similar signal properties are exploited by the prediction and the transform. Nevertheless, performance improvements are still observed.

For the DA-PBT, one of the advantages of being a block transform is that it can be conveniently combined with block-based predictive coding widely adopted in image and video coding standards. However, the block-wise operation is incapable of exploiting the correlation across block boundaries, and additionally, at low rates, results in blocking artifacts in the reconstruction. The DA-DWT proposed for transform coding of still images does not have these drawbacks. Using the lifting structure, the DA-DWT is able to locally adapt the transform directions to the image content while ensuring the reversibility of the transform. Although the transform directions are selected on a block-by-block basis, filtering extends across block boundaries so

that inter-block correlation is exploited and blocking artifacts are absent in the reconstruction. In our experiments, the DA-DWT outperforms the conventional 2-D DWT by around 1 to 2 dB for typical test images, and by up to 5.1 dB for special classes of images, consistent with the performance improvement derived from theoretical analysis using the image model. At low rates, the reconstruction error of the DA-DWT exhibits brushstroke-like artifacts. Different from the ringing and checkerboard artifacts typically observed in the conventional transform, the brushstroke artifacts better preserve the geometric structure in the image. The ability to better depict the image content at low rates, together with embedded coding of the coefficients, makes the DA-DWT especially suitable for progressive transmission of images.

Bibliography

[1] M. D. Adams and F. Kossentini, "Jasper: a software-based JPEG-2000 codec implementation," in *Proc. IEEE Int. Conf. on Image Processing 2000*, Vancouver, Canada, Sept. 2000, vol. 2, pp. 53–56.

[2] N. Ahmed and T. Natarajan K. R. Rao, "Discrete cosine transfom," *IEEE Trans. Comput.*, vol. 23, no. 1, pp. 90–93, Jan. 1974.

[3] A. J. Ahumada, Jr. and H. A. Peterson, "A visual detection model for DCT coefficient quantization," in *Proc. 9th AIAA Computing in Aerospace Conference 1993*, San Diego, CA, Oct. 1993, pp. 314–318.

[4] A. Aiyer, K. Pyun, Y.-Z. Huang, D. B. O'Brien, and R. M. Gray, "Lloyd clustering of Gauss mixture models for image compression and classification," *Signal Processing: Image Communication*, vol. 20, no. 5, pp. 459–485, June 2005.

[5] G. Anderson and T. Huang, "Piecewise fourier transformation for picture bandwidth compression," *IEEE Trans. Commun. Technol.*, vol. 19, no. 2, pp. 133–140, Apr. 1971.

[6] H. C. Andrews and W. K. Pratt, "Fourier transform coding of images," in *Proc. Hawaii Int. Conf. System Science*, Jan. 1968, pp. 677–679.

[7] H. C. Andrews and W. K. Pratt, "Transform image coding," in *Proc. Computer Processing Communications*, New York, NY, USA, 1969, pp. 63–84, Polytechnic Press.

[8] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using vector quantization in the wavelet transform domain," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing 1990*, Albuquerque, NM, USA, Apr. 1990, vol. 4, pp. 2297–2300.

[9] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using wavelet transform," *IEEE Trans. Image Processing*, vol. 1, no. 2, pp. 205–220, Apr. 1992.

[10] R. H. Bamberger and M. J. T. Smith, "A filter bank for the directional decomposition of images: theory and design," *IEEE Trans. Signal Processing*, vol. 40, no. 4, pp. 882–893, Apr. 1992.

[11] I. W. Selesnick R. G. Baraniuk and N. C. Kingsbury, "The dual-tree complex wavelet transform," *IEEE Signal Processing Mag.*, vol. 22, no. 6, pp. 123–151, Nov. 2005.

[12] E. D. Bolker, "The finite Radon transform," *Contemporary Mathematics (American Mathematical Society)*, vol. 63, pp. 27–49, 1987.

[13] N. V. Boulgouris, D. Tzovaras, and M. G. Strintzis, "Lossless image compression based on optimal prediction, adaptive lifting and conditional arithmetic coding," *IEEE Trans. Image Processing*, vol. 10, no. 1, pp. 1–14, Jan. 2001.

[14] J. L. Brown, Jr., "Mean square truncation error in series expansions of random functions," *Journal of the Society for Industrial and Applied Mathematics*, vol. 8, no. 1, pp. 28–32, Mar. 1960.

[15] P. Burt and E. Adelson, "The Laplacian pyramid as a compact image code," *IEEE Trans. Commun.*, vol. 31, no. 4, pp. 532–540, Apr. 1983.

[16] R. Calderbank, I. Daubechies, W. Sweldens, and B.-L. Yeo, "Wavelet transforms that map integers to integers," *Applied and Computational Harmonic Analysis*, vol. 5, no. 3, pp. 332–369, 1998.

[17] E. J. Candès, *Ridgelets: Theory and applications*, Ph.D. dissertation, Stanford University, Stanford, CA, USA, 1998.

[18] E. J. Candès, L. Demanet, D. L. Donoho, and L. Ying, "Fast discrete curvelet transforms," *Multiscale Modeling and Simulation*, vol. 5, no. 3, pp. 861–899, 2006.

[19] E. J. Candès and D. L. Donoho, "Curvelets - a surprisingly effective nonadaptive representation for objects with edges," in *Curves and Surface Fitting: Saint-Malo 1999*, p. 105V120. Vanderbilt Univ. Press, Nashville, TN, USA, 1999.

[20] E. J. Candès and D. L. Donoho, "Ridgelets: a key to higher-dimensional intermittency?," *Phil. Trans. Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 357, no. 1760, pp. 2495–2509, 1999.

[21] E. J. Candès and D. L. Donoho, "New tight frames of curvelets and optimal representations of objects with piecewise c^2 singularities," *Communications on Pure and Applied Mathematics*, vol. 57, no. 2, pp. 219–266, 2004.

[22] E. J. Candès and D. L. Donoho, "Continuous curvelet transform: I. Resolution of the wavefront set," *Appl. Comput. Harmon. Anal.*, vol. 19, pp. 162–197, 2005.

[23] E. J. Candès and D. L. Donoho, "Continuous curvelet transform: II. Discretization and frames," *Appl. Comput. Harmon. Anal.*, vol. 19, pp. 198–222, 2005.

[24] C. Carson, S. Belongie, H. Greenspan, and J. Malik, "Blobworld: Image segmentation using expectation-maximization and its application to image querying," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 24, no. 8, pp. 1026–1038, Aug. 2002.

[25] C.-L. Chang and B. Girod, "Direction-adaptive discrete wavelet transform via directional lifting and bandeletization," in *Proc. IEEE Int. Conf. on Image Processing 2006*, Atlanta, GA, USA, Oct. 2006, pp. 1149–1152.

[26] C.-L. Chang and B. Girod, "Direction-adaptive discrete wavelet transform for image compression," *IEEE Trans. Image Processing*, vol. 16, no. 5, pp. 1289–1302, May 2007.

[27] C.-L. Chang and B. Girod, "Direction-adaptive partitioned block transform for image coding," in *Proc. IEEE Int. Conf. on Image Processing 2008*, San Diego, CA, USA, Oct. 2008, pp. 145–148.

[28] C.-L. Chang and B. Girod, "Image modeling with parametric texture sources for design and analysis of image processing algorithms," in *Proc. SPIE Visual Communications and Image Processing 2008*, San Jose, CA, USA, Jan. 2008, vol. 6822, p. 682229.

[29] C.-L. Chang, M. Makar, S. S. Tsai, and B. Girod, "Direction-adaptive partitioned block transform for color image coding," *(submitted to) IEEE Transactions on Image Processing*, Nov. 2008.

[30] C.-L. Chang, A. Maleki, and B. Girod, "Adaptive wavelet transform for image compression via directional quincunx lifting," in *Proc. IEEE Workshop on Multimedia Signal Processing 2005*, Shanghai, China, Oct. 2005, pp. 1–4.

[31] C.-L. Chang, X. Zhu, P. Ramanathan, and B. Girod, "Inter-view wavelet compression of light fields with disparity-compensated lifting," in *Proc. SPIE Visual Communications and Image Processing 2003*, Lugano, Switzerland, July 2003, vol. 1, pp. 14–22.

[32] C.-L. Chang, X. Zhu, P. Ramanathan, and B. Girod, "Light field compression using disparity-compensated lifting and shape adaptation," *IEEE Trans. Image Processing*, vol. 15, pp. 793–806, Apr. 2006.

[33] C.-T. Chen, "Adaptive transform coding via quadtree-based variable blocksize DCT," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing 1989*, Glasgow, Scotland, May 1989, vol. 3, pp. 1854–1857.

[34] W.-H. Chen, C. Smith, and S. Fralick, "A fast computational algorithm for the discrete cosine transform," *IEEE Trans. Commun.*, vol. 25, no. 9, pp. 1004–1009, Sept. 1977.

[35] P. A. Chou, T. Lookabaugh, and R. M. Gray, "Entropy-constrained vector quantization," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, no. 1, pp. 31–42, Jan. 1989.

[36] C. Christopoulos, A. Skodras, and T. Ebrahimi, "The JPEG2000 still image coding system: an overview," *IEEE Trans. Consumer Electron.*, vol. 46, no. 4, pp. 1103–1127, Nov. 2000.

[37] R. J. Clark, *Transform Coding of Images*, Academic Press, 1985.

[38] R. J. Clarke, "Relation between the Karhunen Loève and cosine transforms," in *IEE Proc. Communications, Radar and Signal Processing*, Nov. 1981, vol. 128, pp. 359–360.

[39] R. Claypoole, G. Davis, W. Sweldens, and R. Baraniuk, "Adaptive wavelet transforms for image coding using lifting," in *Proc. Data Compression Conference 1998*, Snowbird, UT, USA, Mar. 1998, p. 537.

[40] R. L. Claypoole, G. M. Davis, W. Sweldens, and R. G. Baraniuk, "Nonlinear wavelet transforms for image coding via lifting," *IEEE Trans. Image Processing*, vol. 12, no. 12, pp. 1449–1459, Dec. 2003.

[41] W. T. Cochran, J. W. Cooley, D. L. Favin, H. D. Helms, R. A. Kaenel, W. W. Lang, G. C. Maling, Jr., D. E. Nelson, C. M. Rader, and P. D. Welch, "What is the fast fourier transform?," *Proc. IEEE*, vol. 55, no. 10, pp. 1664–1674, Oct. 1967.

[42] A. Cohen, I. Daubechies, and J.-C. Feauveau, "Biorthogonal bases of compactly supported wavelets," *Commun. Pure Appl. Math.*, vol. 45, pp. 485–560, 1992.

[43] J. W. Cooley, P. A. W. Lewis, and P. D. Welch, "Historical notes on the fast fourier transform," *Proc. IEEE*, vol. 55, no. 10, pp. 1675–1677, Oct. 1967.

[44] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, Wiley-Interscience, 1991.

[45] W. R. Crowther and C. M. Rader, "Efficient coding of vocoder channel signals using linear transformation," *Proc. IEEE*, vol. 54, no. 11, pp. 1594–1595, Nov. 1966.

[46] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Commun. Pure Appl. Math.*, vol. 41, pp. 909–996, 1988.

[47] I. Daubechies and W. Sweldens, "Factoring wavelet transforms into lifting steps," *J. Fourier Anal. Appl.*, vol. 4, no. 3, pp. 245–267, 1998.

[48] F. De Simone, M. Ouaret, F. Dufaux, A. G. Tescher, and T. Ebrahimi, "A comparative study of JPEG2000, AVC/H.264, and HD photo," in *Proc. SPIE Applications of Digital Image Processing XXX*, San Diego, CA, USA, 2007, vol. 6696, p. 669602.

[49] R. L. De Valois, D. G. Albrecht, and L. G. Thorell, "Spatial frequency selectivity of cells in macaque visual cortex," *Vision Research*, vol. 22, no. 5, pp. 545–559, 1982.

[50] S. R. Deans, *The Radon Transform and Some of Its Applications*, Wiley, 1983.

[51] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. of the Royal Statistical Soc. Series B*, vol. 39, no. 1, pp. 1–38, 1977.

[52] W. Ding, F. Wu, and S. Li, "Lifting-based wavelet transform with directionally spatial prediction," in *Proc. Picture Coding Symposium 2004*, San Francisco, CA, USA, Dec. 2004, pp. 483–488.

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

[54] I. Dinstein, K. Rose, and A. Heiman, "Variable block-size transform image coder," in *Proc. IEEE Int. Conf. Pattern Recognition 1988*, Rome, Italy, Nov. 1988, vol. 1, pp. 613–617.

[55] I. Dinstein, K. Rose, and A. Heiman, "Variable block-size transform image coder," *IEEE Trans. Commun.*, vol. 38, no. 11, pp. 2073–2078, Nov. 1990.

[56] M. N. Do, *Directional Multiresolution Image Representations*, Ph.D. dissertation, Audio-Visual Communications Laboratory, EPFL, Lausanne, Switzerland, 2001.

[57] M. N. Do, "Contourlets and sparse image expansions," in *Proc. SPIE Wavelets: Applications in Signal and Image Processing X*, San Diego, CA, USA, Nov. 2003, vol. 5207, pp. 560–570.

[58] M. N. Do and M. Vetterli, "Orthonormal finite ridgelet transform for image compression," in *Proc. IEEE Int. Conf. on Image Processing 2000*, Vancouver, Canada, Sept. 2000, vol. 2, pp. 367–370.

[59] M. N. Do and M. Vetterli, "Pyramidal directional filter banks and curvelets," in *Proc. IEEE Int. Conf. on Image Processing 2001*, Thessaloniki, Greece, Oct. 2001, vol. 3, pp. 158–161.

[60] M. N. Do and M. Vetterli, "Contourlets: a directional multiresolution image representation," in *Proc. IEEE Int. Conf. on Image Processing 2002*, Rochester, NY, USA, Sept. 2002, vol. 1, pp. 357–360.

[61] M. N. Do and M. Vetterli, "The finite ridgelet transform for image representation," *IEEE Trans. Image Processing*, vol. 12, no. 1, pp. 16–28, Jan. 2003.

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

[63] D. L. Donoho, "Orthonormal ridgelets and linear singularities," *SIAM Journal on Mathematical Analysis*, vol. 31, no. 5, pp. 1062–1099, 2000.

[64] D. L. Donoho, "Ridge functions and orthonormal ridgelets," *Journal of Approximation Theory*, vol. 111, no. 2, pp. 143–179, 2001.

[65] D. L. Donoho and M. R. Duncan, "Digital curvelet transform: strategy, implementation, and experiments," in *Proc. SPIE Wavelet Applications VII*, Orlando, FL, USA, 2000, vol. 4056, pp. 12–30.

[66] M. Effros and P. A. Chou, "Weighted universal transform coding: universal image compression with the Karhunen-Loève transform," in *Proc. IEEE Int. Conf. on Image Processing 1995*, Los Alamitos, CA, USA, 1995, vol. 2, pp. 61–64.

[67] M. Effros, P. A. Chou, and R. M. Gray, "Weighted universal image compression," *IEEE Trans. Image Processing*, vol. 8, no. 10, pp. 1317–1329, Oct. 1999.

[68] J. E. Eggerton and M. D. Srinath, "Statistical distributions of image DCT coefficients," *Comput. Electr. Eng.*, vol. 12, no. 3-4, pp. 137–145, 1986.

[69] H. Enomoto and K. Shibata, "Orthogonal transform coding system for television signals," *IEEE Trans. Electromagn. Compat.*, vol. 13, no. 3, pp. 11–17, Aug. 1971.

[70] F. C. A. Fernandes, R. L. C. van Spaendonck, and C. S. Burrus, "A new framework for complex wavelet transforms," *IEEE Trans. Signal Processing*, vol. 51, no. 7, pp. 1825–1837, July 2003.

[71] F. C. A. Fernandes, R. L. C. van Spaendonck, and C. S. Burrus, "Multidimensional, mapping-based complex wavelet transforms," *IEEE Trans. Image Processing*, vol. 14, no. 1, pp. 110–124, Jan. 2005.

[72] F. C. A. Fernandes, M. B. Wakin, and R. G. Baraniuk, "Non-redundant, linear-phase, semi-orthogonal, directional complex wavelets," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing 2004*, Montreal, Quebec, Canada, May 2004, vol. 2, pp. 953–956.

[73] M. A. T. Figueiredo and A. K. Jain, "Unsupervised learning of finite mixture models," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 24, no. 3, pp. 381–396, 2002.

[74] J. E. Fowler, "Qccpack: An open-source software library for quantization, compression, and coding," in *Proc. SPIE Applications of Digital Image Processing XXIII*, San Diego, CA, USA, Aug. 2000, pp. 294–301.

[75] W. T. Freeman and E. H. Adelson, "The design and use of steerable filters," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 13, no. 9, pp. 891–906, Sept. 1991.

[76] J. Fu and B. Zeng, "Directional discrete cosine transforms: A theorectical analysis," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing 2007*, Honolulu, Hawaii, USA, Apr. 2007, pp. 1105–1108.

[77] D. Le Gall and A. Tabatabai, "Sub-band coding of digital images using symmetric short kernel filters and arithmetic coding techniques," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing 1988*, New York, NY, USA, Apr. 1988, vol. 2, pp. 761–764.

[78] T. R. Gardos, "H.263+: the new ITU-T Recommendation for video coding at low bit rates," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing 1998*, Seattle, WA, USA, May 1998, vol. 6, pp. 3793–3796.

[79] O. N. Gerek and A. E. Cetin, "Adaptive polyphase subband decomposition structures for image compression," *IEEE Trans. Image Processing*, vol. 9, no. 10, pp. 1649–1660, Oct. 2000.

[80] O. N. Gerek and A. E. Cetin, "A 2-D orientation-adaptive prediction filter in lifting structures for image coding," *IEEE Trans. Image Processing*, vol. 15, no. 1, pp. 106–111, Jan. 2006.

[81] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*, Kluwer, 1991.

[82] B. Girod, "Rate-constrained motion estimation," in *Proc. SPIE Visual Communications and Image Processing 1994*, Chicago, IL, USA, Sept. 1994, pp. 1026–1034.

[83] B. Girod, C.-L. Chang, P. Ramanathan, and X. Zhu, "Light field compression using disparity-compensated lifting," in *Proc. IEEE Int. Conf. on Multimedia and Expo 2003*, Baltimore, MD, USA, July 2003, vol. 1, pp. I–373–6.

[84] B. Girod and S. Han, "Optimum motion-compensated lifting," *IEEE Signal Processing Lett.*, vol. 12, no. 2, pp. 150–153, Feb. 2005.

[85] B. Girod, S. Han, and C.-L. Chang, "Optimum update step for motion-compensated lifted wavelet coding," in *Proc. Picture Coding Symposium 2004*, San Francisco, CA, USA, Dec. 2004.

[86] B. Gold and C. Rader, "The channel vocoder," *IEEE Trans. Audio Electroacoust.*, vol. 15, no. 4, pp. 148–161, Dec. 1967.

[87] A. Gouze, M. Antonini, M. Barlaud, and B. Macq, "Design of signal-adapted multidimensional lifting scheme for lossy coding," *IEEE Trans. Image Processing*, vol. 13, no. 12, pp. 1589–1603, Dec. 2004.

[88] V. K. Goyal, "Theoretical foundations of transform coding," *IEEE Signal Processing Mag.*, vol. 18, no. 5, pp. 9–21, Sept. 2001.

[89] V. K. Goyal, J. Zhuang, and M. Vetterli, "Universal transform coding based on backward adaptation," in *Proc. Data Compression Conference 1997*, Mar. 1997, pp. 231–240.

[90] V. K. Goyal, J. Zhuang, and M. Vetterli, "Transform coding with backward adaptive updates," *IEEE Trans. Inform. Theory*, vol. 46, no. 4, pp. 1623–1633, July 2000.

[91] V. K. Goyal, J. Zhuang, M. Vetterli, and C. Chan, "Transform coding using adaptive bases and quantization," in *Proc. IEEE Int. Conf. on Image Processing 1996*, Lausanne, Switzerland, 1996, vol. 1, pp. 365–368.

[92] R. M. Gray, "Gauss mixture vector quantization," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing 2001*, Salt Lake City, UT, USA, May 2001, vol. 3, pp. 1769–1772.

[93] R. M. Gray, A. Buzo, A. H. Gray, Jr., and Y. Matsuyama, "Distortion measures for speech processing," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 28, no. 4, pp. 367–376, Aug. 1980.

[94] R. M. Gray, A. H. Gray, Jr., G. Rebolledo, and J. Shore, "Rate-distortion speech coding with a minimum discrimination information distortion measure," *IEEE Trans. Inform. Theory*, vol. 27, no. 6, pp. 708–721, Nov. 1981.

[95] R. M. Gray, J. C. Young, and A. K. Aiyer, "Minimum discrimination information clustering: Modeling and quantization with Gauss mixtures," in *Proc. IEEE Int. Conf. on Image Processing 2001*, Thessaloniki, Greece, Oct. 2001, vol. 3, pp. 14–17.

[96] H. Greenspan, S. Belongie, R. Goodman, P. Perona, S. Rakshit, and C. H. Anderson, "Overcomplete steerable pyramid filters and rotation invariance," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition 1994*, Seattle, WA, USA, June 1994, pp. 222–228.

[97] A. Habibi, "Hybrid coding of pictorial data," *IEEE Trans. Commun.*, vol. 22, no. 5, pp. 614–624, May 1974.

[98] A. Habibi and P. A. Wintz, "Image coding by linear transformation and block quantization," *IEEE Trans. Commun. Technol.*, vol. 19, no. 1, pp. 50–62, Feb. 1971.

[99] M. Hamidi and J. Pearl, "Comparison of the cosine and fourier transforms of markov-1 signals," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 24, no. 5, pp. 428–429, Oct. 1976.

[100] J. J. Y. Huang, *Quantization of correlated random variables*, Ph.D. dissertation, Yale University, New Haven, CT, USA, 1962.

[101] J. J. Y. Huang and P. M. Schultheiss, "Block quantization of correlated gaussian random variables," *IEEE Trans. Communications Systems*, vol. 11, no. 3, pp. 289–296, Sept. 1963.

[102] D. H. Hubel and T. N. Wiesel, "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex," *Journal of Physiology*, vol. 160, pp. 106–154, 1962.

[103] D. H. Hubel and T. N. Wiesel, "Receptive fields and functional architecture of monkey striate cortex," *Journal of Physiology*, vol. 195, pp. 215–243, 1968.

[104] A. K. Jain, "Advances in mathematical models for image processing," *Proc. IEEE*, vol. 69, no. 5, pp. 502–528, May 1981.

[105] A. K. Jain, "Image data compression: A review," *Proc. IEEE*, vol. 69, no. 3, pp. 349–389, Mar. 1981.

[106] A. K. Jain, *Foundamentals of digital image processing*, Prentice Hall, 1989.

[107] J. Jain and A. Jain, "Displacement measurement and its application in interframe image coding," *IEEE Trans. Commun.*, vol. 29, no. 12, pp. 1799–1808, Dec. 1981.

[108] G. A. Jones and J. M. Jones, *Elementary Number Theory*, Springer-Verlag, 1998.

[109] JPEG, "ISO/IEC 10918-1 — ITU-T Recommendation T.81, Information Technology: Digital Compression and Coding of Continuous-Tone Still Images," 1992.

[110] JPEG2000, "ISO/IEC 15444-1:2000, Information Technology: JPEG 2000 Image Coding System," 2002.

[111] JVT-X072, *H.264/MPEG-4 AVC Reference Software Manual*, Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, 2007.

[112] P. Kauff and K. Schuur, "A shape-adaptive DCT with block-based DC separation and $\triangle$DC correction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 3, pp. 237–242, June 1998.

[113] N. G. Kingsbury, "The dual-tree complex wavelet transform: A new efficient tool for image restoration and enhancement," in *Proc. European Signal Processing Conf. 1998*, Rhodes, Greece, Sept. 1998, pp. 319–322.

[114] N. G. Kingsbury, "The dual-tree complex wavelet transform: A new technique for shift invariance and directional filters," in *Proc. 8th IEEE DSP Workshop 1998*, Utah, USA, Aug. 1998.

[115] N. G. Kingsbury, "Image processing with complex wavelets," *Philos. Trans. R. Soc. London A, Math. Phys. Sci.*, vol. 357, no. 1760, pp. 2543–2560, Sept. 1999.

[116] N. G. Kingsbury, "A dual-tree complex wavelet transform with improved orthogonality and symmetry properties," in *Proc. IEEE Int. Conf. on Image Processing 2000*, Vancouver, Canada, Sept. 2000, pp. 375–378.

[117] H. Kramer and M. Mathews, "A linear coding for transmitting a set of correlated signals," *IEEE Trans. Inform. Theory*, vol. 2, no. 3, pp. 41–46, Sept. 1956.

[118] E. Y. Lam and J. W. Goodman, "A mathematical analysis of the DCT coefficient distributions for images," *IEEE Trans. Image Processing*, vol. 10, no. 9, pp. 1661–1666, Oct. 2000.

[119] M. H. C. Law, M. A. T. Figueiredo, and A. K. Jain, "Simultaneous feature selection and clustering using mixture models," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 26, no. 9, pp. 1154–1166, Sept. 2004.

[120] B. Lee, "A new algorithm to compute the discrete cosine transform," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 32, no. 6, pp. 1243–1245, Dec. 1984.

[121] A. Leon-Garcia, *Probability and Random Processes for Electrical Engineering, 2nd Edition*, Addison-Wesley, 1994.

[122] J. M. Lervik and T. A. Ramstad, "Optimality of multiple entropy coder systems for nonstationary sources modelled by a mixture distribution," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing 1996*, Atlanta, GA, USA, May 1996, vol. 4, pp. 1874–1877.

[123] H. Lev-Ari, S. R. Parker, and T. Kailath, "Multidimensional maximum-entropy covariance extension," *IEEE Trans. Inform. Theory*, vol. 35, no. 3, pp. 497–508, May 1989.

[124] A. S. Lewis and G. Knowles, "Image compression using the 2-D wavelet transform," *IEEE Trans. Image Processing*, vol. 1, no. 2, pp. 244–250, Apr. 1992.

[125] J. Li, A. Najmi, and R. M. Gray, "Image classification by a two-dimensional hidden markov model," *IEEE Trans. Signal Processing*, vol. 48, no. 2, pp. 517–533, Feb. 2000.

[126] J. Liang and T. D. Tran, "Fast multiplierless approximations of the DCT with the lifting scheme," *IEEE Trans. Signal Processing*, vol. 49, no. 12, pp. 3032–3044, Dec. 2001.

[127] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Commun.*, vol. 28, no. 1, pp. 84–95, Jan. 1980.

[128] P. List, A. Joch, J. Lainema, G. Bjøntegaard, and M. Karczewicz, "Adaptive deblocking filter," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 614–619, July 2003.

[129] Y. Liu and K. N. Ngan, "Weighted adaptive lifting-based wavelet transform for image coding," *IEEE Trans. Image Processing*, vol. 17, no. 4, pp. 500–511, Apr. 2008.

[130] S. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Inform. Theory*, vol. 28, no. 2, pp. 129–137, Mar. 1982.

[131] H. Lohscheller, "A subjectively adapted image communication system," *IEEE Trans. Commun.*, vol. 32, no. 12, pp. 1316–1322, Dec. 1984.

[132] Y. Lu and M. N. Do, "CRISP contourlets: a critically sampled directional multiresolution image representation," in *Proc. SPIE Wavelets: Applications in Signal and Image Processing X*, San Diego, CA, USA, Nov. 2003, vol. 5207, pp. 655–665.

[133] L. Luo, J. Li, S. Li, Z. Zhuang, and Y.-Q. Zhang, "Motion compensated lifting wavelet and its application in video coding," in *Proc. IEEE Int. Conf. on Multimedia and Expo 2001*, Tokyo, Japan, Aug. 2001, pp. 481–484.

[134] S. Mallat, *A wavelet tour of signal processing*, Academic Press, 2 edition, 2002.

[135] S. G. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, no. 7, pp. 674–693, July 1989.

[136] H. S. Malvar, "Lapped transforms for efficient transform/subband coding," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, no. 6, pp. 969–978, June 1990.

[137] H. S. Malvar, "Biorthogonal and nonuniform lapped transforms for transform coding with reduced blocking and ringing artifacts," *IEEE Trans. Signal Processing*, vol. 46, no. 4, pp. 1043–1053, Apr. 1998.

[138] H. S. Malvar and D. H. Staelin, "Reduction of blocking effects in image coding with a lapped orthogonal transform," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing 1988*, New York, NY, USA, Apr. 1988, vol. 2, pp. 781–784.

[139] H. S. Malvar and D. H. Staelin, "The LOT: transform coding without blocking effects," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, no. 4, pp. 553–559, Apr. 1989.

[140] D. Marpe, H. Schwarz, and T. Wiegand, "Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 620–636, July 2003.

[141] D. Marpe, T. Wiegand, and S. Gordon, "H.264/MPEG4-AVC fidelity range extensions: tools, profiles, performance, and application areas," in *Proc. IEEE Int. Conf. on Image Processing 2005*, Genova, Italy, Sept. 2005, pp. 593–596.

[142] F. Matúš and J. Flusser, "Image representation via a finite Radon transform," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 15, no. 10, pp. 996–1006, Oct. 1993.

[143] A. N. Netravali and J. O. Limb, "Picture coding: A review," *Proc. IEEE*, vol. 68, no. 3, pp. 366–406, Mar. 1980.

[144] T. T. Nguyen and S. Oraintara, "A directional decomposition: theory, design, and implementation," in *Proc. IEEE Int. Symp. Circuits and Systems 2004*, Vancouver, Canada, May 2004, vol. 3, pp. 281–284.

[145] J.-R. Ohm, "Three-dimensional subband coding with motion compensation," *IEEE Trans. Image Processing*, vol. 3, no. 5, pp. 559–571, Sept. 1994.

[146] B. A. Olshausen and D. J. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, pp. 607–609, 1996.

[147] J. B. O'Neal, Jr. and T. R. Natarajan, "Coding isotropic images," *IEEE Trans. Inform. Theory*, vol. 23, no. 6, pp. 697–707, Nov. 1977.

[148] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*, Prentice Hall, 2 edition, 1998.

[149] A. J. Parker and M. J. Hawken, "Two-dimensional spatial structure of receptive fields in monkey striate cortex," *Journal of the Optical Society of America. A, Optics and Image Science*, vol. 5, no. 4, pp. 598–605, Apr. 1988.

[150] W. B. Pennebaker and J. L. Mitchell, *JPEG Still Image Data Compression Standard*, Van Nostrand Reinhold, 1993.

[151] E. L. Pennec and S. Mallat, "Sparse geometric image representations with bandelets," *IEEE Trans. Image Processing*, vol. 14, no. 4, pp. 423–438, Apr. 2005.

[152] B. Pesquet-Popescu and V. Bottreau, "Three dimensional lifting schemes for motion compensated video compression," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing 2001*, Salt Lake City, UT, USA, May 2001, vol. 3, pp. 1793–1796.

[153] G. Peyre and S. Mallat, "Discrete bandelets with geometric orthogonal filters," in *Proc. IEEE Int. Conf. on Image Processing 2005*, Genova, Italy, Sept. 2005, pp. I–65–8.

[154] G. Peyre and S. Mallat, "Surface compression with geometric bandelets," *ACM Transactions on Graphics*, vol. 24, no. 3, pp. 601–608, July 2005.

[155] W. K. Pratt, W.-H. Chen, and L. Welch, "Slant transform image coding," *IEEE Trans. Commun.*, vol. 22, no. 8, pp. 1075–1093, Aug. 1974.

[156] W. K. Pratt, J. Kane, and H. C. Andrews, "Hadamard transform image coding," *Proc. IEEE*, vol. 57, no. 1, pp. 58–68, Jan. 1969.

[157] B. Ramamurthi and A. Gersho, "Classified vector quantization of images," *IEEE Trans. Commun.*, vol. 34, no. 11, pp. 1105–1115, Nov. 1986.

[158] K. Ramchandran and M. Vetterli, "Best wavelet packet bases in a rate-distortion sense," *IEEE Trans. Image Processing*, vol. 2, no. 2, pp. 160–175, Apr. 1993.

[159] K. Ramchandran, Z. Xiong, K. Asai, and M. Vetterli, "Adaptive transforms for image coding using spatially varying wavelet packets," *IEEE Trans. Image Processing*, vol. 5, no. 7, pp. 1197–1204, July 1996.

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

[161] R. Reininger and J. Gibson, "Distributions of the two-dimensional DCT coefficients for images," *IEEE Trans. Commun.*, vol. 31, no. 6, pp. 835–839, June 1983.

[162] K. Rijkse, "H.263: video coding for low-bit-rate communication," *IEEE Commun. Mag.*, vol. 34, no. 12, pp. 42–45, Dec. 1996.

[163] A. Robert, I. Amonou, and B. Pesquet-Popescu, "Improving intra mode coding in H.264/AVC through block oriented transforms," in *Proc. IEEE Multimedia Signal Processing 2006*, Victoria, BC, Canada, Oct. 2006, pp. 382–386.

[164] M. A. Robertson and R. L. Stevenson, "DCT quantization noise in compressed images," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 1, pp. 27–38, Jan. 2005.

[165] J. Roese, W. Pratt, and G. Robinson, "Interframe cosine transform image coding," *IEEE Trans. Commun.*, vol. 25, no. 11, pp. 1329–1339, Nov. 1977.

[166] A. Said and W. A. Pearlman, "A new fast and efficient image codec based on Set Partitioning in Hierarchical Trees," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 3, pp. 243–250, June 1996.

[167] D. J. Sakrison and V. R. Algazi, "Comparison of line-by-line and two-dimensional encodings of random images," *IEEE Trans. Inform. Theory*, vol. 17, no. 4, pp. 386–398, July 1971.

[168] A. Secker and D. Taubman, "Motion-compensated highly scalable video compression using an adaptive 3D wavelet transform based on lifting," in *Proc. IEEE Int. Conf. on Image Processing 2001*, Thessaloniki, Greece, Oct. 2001, vol. 2, pp. 1029–1032.

[169] A. Secker and D. Taubman, "Highly scalable video compression with scalable motion coding," *IEEE Trans. Image Processing*, vol. 13, pp. 1029–1041, Aug. 2004.

[170] V. P. Shah, J. E. Fowler, and N. H. Younan, "Tarp filtering of block-transform coefficients for embedded image coding," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing 2006*, Toulouse, France, May 2006, pp. II–II.

[171] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. Signal Processing*, vol. 41, no. 12, pp. 3445–3462, Dec. 1993.

[172] M.-Y. Shen and C.-C. J. Kuo, "Review of postprocessing techniques for compression artifact removal," *Journal of Visual Communication and Image Representation*, vol. 9, no. 1, pp. 2–14, 1998.

[173] P. Simard, D. Steinkraus, and H. Malvar, "On-line adaptation in image coding with a 2-D tarp filter," in *Proc. Data Compression Conference 2002*, Snowbird, UT, USA, May 2002, pp. 23–32.

[174] E. P. Simoncelli, W. T. Freeman, E. H. Adelson, and D. H. Heeger, "Shiftable multiscale transforms," *IEEE Trans. Inform. Theory*, vol. 38, no. 2, pp. 587–607, Mar. 1992.

[175] A. Skodras, C. Christopoulos, and T. Ebrahimi, "The JPEG 2000 still image compression standard," *IEEE Signal Processing Mag.*, vol. 18, no. 5, pp. 36–58, Sept. 2001.

[176] S. Srinivasan, C. Tu, S. L. Regunathan, and G. J. Sullivan, "HD Photo: a new image coding technology for digital photography," in *Proc. SPIE Applications of Digital Image Processing XXX*, San Diego, CA, USA, 2007, vol. 6696, p. 66960A.

[177] A. Sripad and D. Snyder, "A necessary and sufficient condition for quantization errors to be uniform and white," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 25, no. 5, pp. 442–448, Oct. 1977.

[178] J.-L. Starck, E. J. Candès, and D. L. Donoho, "The curvelet transform for image denoising," *IEEE Trans. Image Processing*, vol. 11, no. 6, pp. 670–684, June 2002.

[179] W. Sweldens, "The lifting scheme: A construction of second generation wavelets," *SIAM Journal on Mathematical Analysis*, vol. 29, no. 2, pp. 511–546, 1998.

[180] M. Tasto and P. A. Wintz, "Image coding by adaptive block quantization," *IEEE Trans. Commun. Technol.*, vol. 19, no. 6, pp. 957–972, Dec. 1971.

[181] M. Tasto and P. A. Wintz, "A bound on the rate-distortion function and application to images," *IEEE Trans. Inform. Theory*, vol. 18, no. 1, pp. 150–159, Jan. 1972.

[182] D. Taubman, "Adaptive, non-separable lifting transforms for image compression," in *Proc. IEEE Int. Conf. on Image Processing 1999*, Kobe, Japan, Oct. 1999, vol. 3, pp. 772–776.

[183] D. Taubman, "High performance scalable image compression with EBCOT," *IEEE Trans. Image Processing*, vol. 9, no. 7, pp. 1158–1170, July 2000.

[184] D. Taubman and A. Zakhor, "Orientation adaptive subband coding of images," *IEEE Trans. Image Processing*, vol. 3, no. 4, pp. 421–437, July 1994.

[185] D. S. Taubman and M. W. Marcellin, *JPEG2000: Image Compression Fundamentals, Standards and Practice*, Kluwer Academic Publishers, 2002.

[186] C. Tian and S. S. Hemami, "An embedded image coding system based on tarp filter with classification," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing 2004*, Montreal, Quebec, Canada, May 2004, vol. 3, pp. 49–52.

[187] T. D. Tran, J. Liang, and C. Tu, "Lapped transform via time-domain pre- and post-filtering," *IEEE Trans. Signal Processing*, vol. 51, no. 6, pp. 1557–1571, June 2003.

[188] B. E. Usevitch, "A tutorial on modern lossy wavelet image compression: foundations of JPEG 2000," *IEEE Signal Processing Mag.*, vol. 18, no. 5, pp. 22–35, Sept. 2001.

[189] D. Vaisey and A. Gersho, "Variable block-size image coding," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing 1987*, Dallas, TX, USA, Apr. 1987, vol. 12, pp. 1051–1054.

[190] J. Vaisey and A. Gersho, "Image compression with variable block size segmentation," *IEEE Trans. Signal Processing*, vol. 40, no. 8, pp. 2040–2060, Aug. 1992.

[191] V. Velisavljevic, *Directionlets: anisotropic multi-directional representation with separable filtering*, Ph.D. dissertation, School of Computer and Communication Sciences, EPFL, Lausanne, Switzerland, 2005.

[192] V. Velisavljevic, B. Beferull-Lozano, and M. Vetterli, "Space-frequency quantization for image compression with directionlets," *IEEE Trans. Image Processing*, vol. 16, no. 7, pp. 1761–1773, July 2007.

[193] V. Velisavljevic, B. Beferull-Lozano, M. Vetterli, and P. L. Dragotti, "Approximation power of directionlets," in *Proc. IEEE Int. Conf. on Image Processing 2005*, Genova, Italy, Sept. 2005, vol. 1, pp. 741–744.

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

[195] M. Vetterli, "Multi-dimensional subband coding: some theory and algorithms," *Signal Processing*, vol. 6, no. 2, pp. 97–112, 1984.

[196] M. Vetterli, "Wavelets, approximation, and compression," *IEEE Signal Processing Mag.*, vol. 18, no. 5, pp. 59–73, Sept. 2001.

[197] M. Vetterli and J. Kovacevic, *Wavelets and Subband Coding*, Prentice-Hall, 1995.

[198] G. K. Wallace, "Overview of the JPEG (ISO/CCITT) still image compression standard," in *Proc. SPIE Image Processing Algorithms and Techniques*, Santa Clara, CA, USA, 1990, vol. 1244, pp. 220–233.

[199] G. K. Wallace, "The JPEG still picture compression standard," *IEEE Trans. Consumer Electron.*, vol. 38, no. 1, pp. xviii–xxxiv, Feb. 1992.

[200] D. Wang, L. Zhang, and A. Vincent, "Improvement of JPEG2000 using curved wavelet transform," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing 2005*, Philadelphia, PA, USA, Mar. 2005, vol. 2, pp. 365–368.

[201] D. Wang, L. Zhang, A. Vincent, and F. Speranza, "Curved wavelet transform for image coding," *IEEE Trans. Image Processing*, vol. 15, no. 8, pp. 2413–2421, Aug. 2006.

[202] T. Wedi and S. Wittmann, "Quantization offsets for video coding," in *Proc. IEEE Int. Symp. on Circuits and Systems 2005*, Kobe, Japan, May 2005, pp. 324–327.

[203] P. H. Westerink, D. E. Boekee, J. Biemond, and J. W. Woods, "Subband coding of images using vector quantization," *IEEE Trans. Commun.*, vol. 36, no. 6, pp. 713–719, June 1988.

[204] T. Wiegand and B. Girod, "Lagrange multiplier selection in hybrid video coder control," in *Proc. IEEE Int. Conf. on Image Processing 2001*, Thessaloniki, Greece, Oct. 2001, vol. 3, pp. 542–545.

[205] T. Wiegand, H. Schwarz, A. Joch, and F. Kossentini, "Rate-constrained coder control and comparison of video coding standards," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 688–703, July 2003.

[206] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, July 2003.

[207] M. Wien, "Variable block-size transforms for H.264/AVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 604–613, July 2003.

[208] P. A. Wintz, "Transform picture coding," *Proc. IEEE*, vol. 60, no. 7, pp. 809–820, July 1972.

[209] J. Woods and S. O'Neil, "Subband coding of images," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 34, no. 5, pp. 1278–1288, Oct. 1986.

[210] R. Xiong, F. Wu, J. Xu, S. Li, and Y.-Q. Zhang, "Barbell lifting wavelet transform for highly scalable video coding," in *Proc. Picture Coding Symposium 2004*, San Francisco, CA, USA, Dec. 2004, pp. 237–242.

[211] R. Xiong, J. Xu, F. Wu, and S. Li, "Barbell-lifting based 3-D wavelet coding scheme," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1256–1269, Sept. 2007.

[212] R. Xiong, J. Xu, F. Wu, S. Li, and Y.-Q. Zhang, "Layered motion estimation and coding for fully scalable 3D wavelet video coding," in *Proc. IEEE Int. Conf. on Image Processing 2004*, Singapore, Oct. 2004, pp. 2271–2274.

[213] Z. Xiong, O. G. Guleryuz, and M. T. Orchard, "A DCT-based embedded image coder," *IEEE Signal Processing Letters*, vol. 3, no. 11, pp. 289–290, Nov. 1996.

[214] H. Xu, J. Xu, and F. Wu, "Lifting-based directional DCT-like transform for image coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 10, pp. 1325–1335, Oct. 2007.

[215] J. Xu, F. Wu, J. Liang, and W. Zhang, "Directional lapped transforms for image coding," in *Proc. Data Compression Conference 2008*, Snowbird, UT, USA, Mar. 2008, pp. 142–151.

[216] T. Xu, C.-L. Chang, and B. Girod, "Scalable direction representation for image compression with direction-adaptive discrete wavelet transform," in *Proc. SPIE Visual Communications and Image Processing 2007*, San Jose, CA, USA, Jan. 2007, vol. 6508, p. 65080V.

[217] Y. Ye and M. Karczewicz, "Improved H.264 intra coding based on bi-directional intra prediction, directional transform, and adaptive coefficient scanning," in *Proc. IEEE Int. Conf. on Image Processing 2008*, San Diego, CA, USA, Oct. 2008, pp. 2116–2119.

[218] B. Zeng and J. Fu, "Directional discrete cosine transforms for image coding," in *Proc. IEEE Int. Conf. on Multimedia and Expo 2006*, Toronto, Ontario, Canada, July 2006, pp. 721–724.

[219] B. Zeng and J. Fu, "Directional discrete cosine transforms - a new framework for image coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 3, pp. 305–313, Mar. 2008.

[220] J. Zhang and D. Ma, "Nonlinear prediction for gaussian mixture image models," *IEEE Trans. Image Processing*, vol. 13, no. 6, pp. 836–847, June 2004.

[221] S. Zhang and C. Moloney, "The nonredundant contourlet transform (NRCT): A multiresolution and multidirection image representation," in *Proc. Canadian Conf. Elec. and Computer Engineering 2008*, May 2008, pp. 1323–1326.