Perceptual image coding and transcoding with subband discrete cosine transform

Tan, Ee Leng

2011

Tan, E. L. (2011). Perceptual image coding and transcoding with subband discrete cosine transform. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/48915>

<https://doi.org/10.32657/10356/48915>

Perceptual Image Coding and Transcoding with Subband Discrete Cosine Transform

Tan Ee Leng

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirement for the degree of
Doctor of Philosophy

2011

Acknowledgements

First and foremost, I would like to express my greatest gratitude to my supervisor, Assoc. Prof. Gan Woon Seng, for his invaluable guidance and encouragement throughout the course of this work. I would like to thank him for the opportunity to perform my PhD work in the Digital Signal Processing Lab. The unique flavor of this lab was always motivating. I also owe a special thanks to Mrs Gan who persuaded me to pursue a PhD degree, when I was approached by Prof. Gan.

I sincerely thank Dr. Lee Kong Aik, Dr. Wang Liang, Dr. Wen Yuan, and Mr. Furi Andi Karnapi for all the valuable chats, and they have always been encouraging and supportive. I would also like to thank all my friends from the Lab, Jiwei, Shi Chuang, Tong Wei, Mu Hao, Abhishek, Dileep, Kushal, Nay Oo, Reuben, Zhang Hua, and Dahyanto. They have made the lab an interesting place for doing research and study. I am also grateful to technicians, Mr. Yeo Sung Kheng and Mr. Ong Say Cheng, for their continual support and help in logistic and administrative matters. I would like to thank all the friends and colleagues of the lab who participated in the subjective testing, and providing me with useful feedback and comments that are instrumental in completing this thesis.

Finally, I would like to thank my family members who constantly encouraged and supported me throughout the course of my PhD. I dedicate this thesis to them.

Ee Leng Tan

Singapore, 2011

Table of Contents

Acknowledgements	ii
Table of Contents	iii
Summary	vi
List of Figures	ix
List of Tables	xiv
List of Abbreviations and Acronyms	xvi
List of Symbols	xviii
Chapter 1 Introduction	1
1.1 Anatomy of the Early Human Visual System	3
1.2 Motivation of Thesis	8
1.3 Contribution of Thesis	9
1.4 Thesis Overview	11
Chapter 2 Computational Models for Just Noticeable Differences	14
2.1 Frequency Decomposition	18
2.1.1 Cortex Filters	19
2.1.2 Mapping DCT Coefficients to Cortex Bands	22
2.2 Spatial Contrast Sensitivity Function	23
2.2.1 Determining Luminance Dependent Parameters of Spatial CSF	27
2.3 Luminance Adaptation	29
2.4 Contrast Masking	31

2.4.1	Estimation of Inter-Band Contrast Masking from Cortex Bands	37
2.5	Error Pooling	38
2.6	Conversion between JND Models from Subband and Pixel	40
2.7	Conclusions	41
Chapter 3	Subband Discrete Cosine Transform	44
3.1	Derivation of Subband Discrete Cosine Transform	45
3.2	Fast Subband Discrete Cosine Transform	48
3.2.1	Fast 4×4 DCT-II Algorithm	56
3.2.2	Computational Cost Analysis of Fast Subband Discrete Cosine Transform and Approximations of DCT-II	63
3.3	Adaptive Subband Discrete Cosine Transform	69
3.4	Simulation Results and Discussion	79
3.5	Conclusions	88
Chapter 4	Perceptual Image Coding with SBDCT and JND Modeling	92
4.1	Proposed Encoder	94
4.1.1	Spatial Just Noticeable Difference Model	98
4.1.2	Block Classification of $N \times N$ Sub-Image	99
4.1.3	Decomposition of JND Profile of an Image	104
4.1.4	Computing Base Detection Threshold from MTF	106
4.1.5	Integrating SBDCT and JND Model	109
4.2	Verification of JND Modeling	114
4.2.1	Error Pooling	127

4.3	Performance Comparison of PICs	134
4.4	Conclusions	141
Chapter 5	Perceptual Image Transcoding with SBDCT	144
5.1	Spatial Relationship of DCT-II Coefficients Between a Block and its Sub-Blocks	147
5.1.1	Generalization of Symmetry Property of Spatial Relationship of DCT-II Coefficients Between a Block and its Sub-Blocks	149
5.2	Image Transcoding Method to Resize Image	153
5.2.1	Resizing Images by a Rational Factor of P/Q	153
5.2.2	Reduction of Computational Cost By Sparse Matrix Representation	156
5.2.3	Reduction of Computational Cost Using a Sub-Set of Up-Sampled DCT-II Coefficients	158
5.3	Perceptual Image Transcoding	173
5.4	Results and Discussion	179
5.5	Conclusions	187
Chapter 6	Conclusions and Future Works	187
6.1	Conclusions	187
6.2	Future Works	197
Author's Publications		199
Bibliography		201
Appendix A - Derivation of aliased components in subband discrete cosine transform		219
Appendix B - Test - images		226
Appendix C - PSNR and MSSIM values		231

Summary

One of the biggest challenges in the field of image and video processing is evaluating and optimizing the quality of digital imaging system with respect to storage and transmission of visual information. Due to physiological and psychological mechanisms of the human visual system (HVS), the HVS is unable to detect all changes in an image. By exploiting limitations of the HVS, scarce resources of digital imaging system such as storage space as well as transmission bandwidth can be optimally allocated for the optimal visual experience.

Discrete cosine transform (DCT) is used in many image and video standards (JPEG, MPEG-1/2/4, H.261/3), therefore this thesis focuses on a computational model to determine the threshold of just-noticeable-difference (JND) based on DCT subband. Several well accepted factors such as contrast sensitivity function (CSF), luminance adaptation and contrast masking, which influence the visibility of differences (or distortion) in an image are investigated. Specifically, these visual factors are incorporated in a computational model for JND to guide an image compression algorithm in achieving higher compression ratio without introducing perceivable distortion.

A variant of the Type-II DCT (DCT-II) namely the subband discrete cosine transform (SBDCT) has been applied in this thesis. One of the main differences between the two-dimensional (2-D) DCT-II and 2-D SBDCT is the 2-D DCT-II and 2-D SBDCT operate on an $N \times N$ input sequence and subbands of an $N \times N$ input sequence, respectively. In the case of SBDCT, the subbands of an $N \times N$ input sequence can be independently processed, thus, SBDCT is well-suited for parallel implementation. Furthermore, it has been shown that significant reduction of computational cost is achieved by approximating the 2-D DCT-II using the

SBDCT in a image compression system, while maintaining reasonable image quality in the reconstructed image.

The first part of this thesis discusses on a fast computational structure for the SBDCT. Based on a symmetry property found in the SBDCT, a fast computational structure for the SBDCT is derived. Followed by, seven approximations of the 2-D DCT-II based on the SBDCT are obtained. The second part of this thesis focuses on the development of an image compression system, which integrates the SBDCT with a computational model for JND. A computational model for JND is adapted to the SBDCT, which takes into account the computational structure of SBDCT to obtain an effective image compression system. The proposed image compression system is subjected to a series of subjective evaluations to validate its effectiveness in terms of compression ratio and perceived image quality of the reconstructed image.

With the emergence of mobile multimedia capable devices and available of broadband networks, users have easy access to high definition image and video contents. Hence, an efficient method to reduce the spatial resolution of an image or video content before delivery to such mobile devices is desired. The third and the final part of this thesis presents a computationally efficient image transcoding algorithm to resize images, which does not require any DCT or inverse DCT (IDCT) algorithm employed in many existing image resizing algorithms. Instead, the proposed algorithm applies the spatial relationship of the DCT-II coefficients between a block and its sub-blocks as well as the observation of the SBDCT to obtain an efficient computational structure.

Generally, arbitrarily resizing of images in the DCT domain is achieved by first up-sampling and followed by down-sampling of DCT-II coefficients. To

reduce computational cost without degrading visual quality, the proposed transcoding algorithm utilizes a subset of the up-sampled DCT coefficients, instead of all the up-sampled DCT-II coefficients, in the down-sampling operation. The selection of the up-sampled DCT-II coefficients is obtained empirically through the analysis of the image degradation with respect to the number of up-sampled DCT-II coefficients used in the down-sampling process.

To further reduce computational cost of the proposed image transcoding algorithm, the base visibility threshold derived in the computational model for JND is used to determine the number of up-sampled DCT-II coefficients used in the down-sampling process. Subjective evaluation of the resized images confirms that no perceptible distortion is introduced in the resized images using this JND guided image transcoding algorithm, as compared to the resized images obtained with all the up-sampled DCT-II coefficients in the down-sampling operation.

List of Figures

Figure 1.1	Four main processing blocks of the early HVS consisting of the optics of the eye, retina, LGN, and visual cortex. Adapted from [ZB06].	4
Figure 1.2	Optics of the eye.	5
Figure 1.3	Visual angle of the eye.	5
Figure 1.4	Point spread function of the human eye as a function of distance [Wes86].	6
Figure 1.5	Density of cones and rods as a function of eccentricity. Adapted from [Ost35].	7
Figure 2.1	Block diagram of PIC.	17
Figure 2.2	Decomposition of the frequency plane corresponding to (a) Watson's cortex transform [Wat87], (b) Daly's cortex filters [Dal92, Dal93], and (c) DCT-II. The range of each axis is from $-f_s/2$ to $f_s/2$, where f_s is the sampling frequency.	19
Figure 2.3	Frequency response of a mesa filter [Dal92, Dal93].	20
Figure 2.4	Mapping of DCT coefficients (thin line) to cortex bands (thick line). The shaded area denotes the DCT coefficients which fall within the same cortex band. Dashed lines denotes the sub-bins of each DCT coefficient.	23
Figure 2.5	Contrast sensitivity data from van Nes and Bouman [NB67].	26
Figure 2.6	CSFs at various mean luminance levels. Plots are calculated using the empirical CSF model in [Bar04]. The region where Weber's law holds is circled.	30
Figure 2.7	Operator to determine average local luminance (B).	32
Figure 2.8	Contrast threshold for detecting a 2 cpd grating (target) as a function of the contrast of a masking grating ranging from 1 to 4 cpd. Extracted from [LF80].	33
Figure 2.9	DCT-II block classification for contrast masking. LF, MF, and HF are represented by the dark gray, light gray and white boxes, respectively.	35
Figure 2.10	Operators to determine weighted average of luminance changes (G_p).	38

Figure 3.1	Four subband transforms of $c(b_i, \mathbf{q}, \mathbf{n})$ in SBDCT.	48
Figure 3.2	Computational structure of 4-point DCT-II. The computational cost of this structure is (4, 9).	62
Figure 3.3	Computational structure of 4×4 DCT-II. The computational cost of this structure is (16, 74) and 9 right shifts.	62
Figure 3.4	Signal flow diagram of N' -point subband DCT. For a 4-point input, $C(0, 4, n_1)$ is 0 and this computational structure requires 3 multiplications and 9 additions.	65
Figure 3.5	Signal flow diagram of N' -point subband DST. For a 4-point input, $C(1, 0, n_1)$ is 0 and this computational structure requires 4 multiplications and 9 additions.	65
Figure 3.6	Mean square error between $\tilde{\mathbf{C}}_{n_1}^{1D}$ and DCT-II.	70
Figure 3.7	Total alias of MSA_1 to MSA_7 for a block size 2^N .	77
Figure 3.8	Scatter plot of PSNR versus MSSIM of the images reconstructed using AdS1. The PSNR and MSSIM values are presented in Table 3.10.	82
Figure 3.9	Reconstructing TI-25 using AdS1, (a) original image, (b) reconstructed image, (c) SSIM map, and (d) error map. These cropped images are un-scaled.	84
Figure 3.10	Scatter plot of PSNR versus MSSIM of the images reconstructed using AdS2. The PSNR and MSSIM values are presented in Table 3.11.	87
Figure 3.11	Cropped images of original image (left), reconstructed image using AdS1 (middle), reconstructed image using AdS2 (right) for (a) TI-1 and (b) TI-25. These cropped images are un-scaled.	87
Figure 3.12	Test images of (a) TI-8 and (b) TI-19, and SSIM maps of reconstructed images of (c) TI-8 using AdS1, (d) TI-19 using AdS1, (e) TI-8 using AdS2, and (f) TI-19 using AdS2.	89
Figure 3.13	Error and SSIM maps of reconstructed images of TI-21. (a) Original image of TI-8. (b) SSIM maps of reconstructed images of TI-21 using AdS1 and AdS2 are (b) and (c), respectively. Error maps of reconstructed images of TI-21 using AdS1 and AdS2 are (d) and (e), respectively	90
Figure 4.1	Block diagram of the proposed PIC.	96

Figure 4.2	Edge map of "Barbara" and "Baboon" images obtained using (a) to (b) subbands $\mathbf{c}_{1,n}$, $\mathbf{c}_{2,n}$, and $\mathbf{c}_{3,n}$, (c) to (d) subbands $\mathbf{c}_{1,n}$, $\mathbf{c}_{2,n}$, and $\mathbf{c}_{3,n}$ with thresholding, and (e) to (f) Canny edge operator with threshold set to 0.1.	102
Figure 4.3	Block classification results of "Barbara" and "Baboon" images using (a) to (b) Wei's method, (c) to (d) Zhang's method, and (e) to (f) proposed method. Black, gray and white regions indicate plain, texture and edge regions, respectively.	103
Figure 4.4	Comparison of various MTFs for $N = 8$.	108
Figure 4.5	Segments of the "Lena" image extracted from the constructed image, which are compressed using (a) DCT-II, (b) ApD1, and (c) PSBDCT and then reconstructed by IDCT-II decoder.	110
Figure 4.6	Signal flow diagram to decompose signal $c(p_1, n_1)$ into two sub-sequences $c(0, q_1, n_1)$ and $c(1, q_1, n_1)$.	111
Figure 4.7	Frequency response of $h_a(m)$ and $h_b(m)$.	111
Figure 4.8	JND profiles of TI- 6 with horizontal grating at 4 cpd. JND profiles based on t_{Wat} , t_{CL} , t_{YLL} , t_{ZLX} , t_{WN} , and t_{prop} are shown in (a) to (g), respectively.	118
Figure 4.9	JND profiles of TI-6 with horizontal grating at 20 cpd. JND profiles based on t_{Wat} , t_{CL} , t_{YLL} , t_{ZLX} , t_{WN} , and t_{prop} are shown in (a) to (g), respectively.	119
Figure 4.10	JND profiles of TI-10 with horizontal grating at 4 (left image) and 20 (right image) cpd. JND profiles based on t_{Wat} , t_{CL} , t_{YLL} , t_{ZLX} , t_{WN} , and t_{prop} are shown in (a) to (g), respectively.	121
Figure 4.12	Subjective rating and its associated 95% confidence interval of noise-contaminated images (PSNR = 30 dB) from various JND models.	123
Figure 4.12	Noise-contaminated images of TI-8 from various JND models, where (a) is the cropped original image. Noise-contaminated images from t_{Wat} , t_{CL} , t_{YLL} , t_{ZLX} , t_{WN} , and t_{prop} are (b) to (g), respectively.	124

Figure 4.13	Noise-contaminated images obtained from DCT subband-based JND models. (a) Original TI-17, (b) original TI-20, (c) to (d) Watson's model, (e) to (f) Zhang's model, (g) to (h) Wei and Ngan's model, and (i) to (j) Proposed model.	126
Figure 4.14	Subjective rating and its associated 95% confidence interval of noise-contaminated images at various bit-rates and values of β in error pooling of the proposed JND model.	130
Figure 4.15	Scatter plots of subjective rating versus perceptual distortion score. Fitted polynomial function is shown in red. Scatter plots of proposed PIC using β_1 , β_2 , and β_3 are shown in (a), (b), and (c), respectively.	133
Figure 4.16	Subjective rating and its associated 95% confidence interval of noise-contaminated images for various values of β in error pooling of the proposed JND model.	134
Figure 4.17	Subjective rating and its associated 95% confidence interval of noise-contaminated images from various PICs	136
Figure 4.18	Scatter plots of subjective rating vs. subjective rating. Fitted polynomial function is shown in red. Scatter plots of Watson's, Wei's, Zhang's and the proposed PICs are shown in (a), (b), (c), and (d), respectively.	139
Figure 4.19	Pearson and Spearman rank-order correlations of the four PIC and its associated 95% confidence interval	140
Figure 5.1	Relation between sub-matrices of $\mathbf{A}_{(Q,N)}$. Entries of $\mathbf{D}_{l_1, Q-l_2-1}$ can be obtained as the product of the corresponding entries in the relationship table and $\mathbf{D}_{l_1, l_2}$. The light and dark gray boxes in the relationship table represent 1 and -1 , respectively.	151
Figure 5.2	Proposed method to resize a 1-D signal by a factor of P/Q using up-sampled DCT-II blocks each of size N . (a) Q N -point DCT-II blocks are used, (b) to (c) each DCT-II block is up-sampled by a factor of P , and (d) to (e) P consecutive Q up-sampled DCT-II blocks are down-sampled by a factor of Q .	155
Figure 5.3	Faster proposed method to resize a 1-D signal by a factor of P/Q using a block size $\tilde{N}$ for each up-sampled DCT-II block. (a) Q N -point DCT-II blocks are used, (b) – (c) each DCT-II block is up-sampled by a factor of P , (d) the last $N - \tilde{N}$ up-sampled coefficients for each N -point DCT-II block are zeroed, (e) – (f) P sets of Q consecutive up-sampled DCT-II blocks are down-sampled by a factor of Q .	159

Figure 5.4	A subset of up-sampled DCT-II coefficients considered in 1-D resizing. (a) Computational cost of the up-sampling stage is reduced using $\hat{\mathbf{U}}_{u_1,0}$, where the bottom $(N \text{ to } \tilde{N}) \times N$ entries are zero and (b) computational cost of the down-sampling stage is reduced using $\hat{\mathbf{D}}_{0,d_1}$, where the right $N \times (N - \tilde{N})$ entries are zero.	161
Figure 5.5	Filterbank structure of the down-sampling stage in the proposed method. N decimation filters are specified by the decimation filter matrix $\mathbf{H}$, h_m represents the m th row of $\mathbf{H}$, and $n = 0, 1, \dots, N - 1$.	162
Figure 5.6	Magnitude response of equivalent filter defined by $\mathbf{H}$ ($N = 8$) and $\hat{\mathbf{H}}$ (a) for down-sampling by a factor of three and (b) for down-sampling by a factor of four.	164
Figure 5.7	Downsizing and upsizing of images for image comparison using PSNR and MSSIM.	168
Figure 5.8	Average differential PSNR and MSSIM of test images down-sampled by a factor of Q . (a) Average differential PSNR for $Q = 2, 3, \dots, 7$ and (b) average differential PSNR for $Q = 8, 9, \dots, 13$. (c) Average differential MSSIM for $Q = 2, 3, \dots, 7$ and (d) average differential MSSIM for $Q = 8, 9, \dots, 13$.	169
Figure 5.9	Average PSNR difference for various downsizing factors for $Q = 3$ to 7 in (a) to (e), respectively.	171
Figure 5.10	Average PSNR difference for various downsizing factors for $Q = 8$ to 12 in (a) to (e), respectively.	172
Figure 5.11	Perceptual distortion score and its associated 95% confidence interval of images resized by a factor of P/Q .	178
Figure 5.12	Generalized operations of the methods proposed by Park [PP06] and Shu [SC07].	180
Figure 5.13	High definition grayscale test-images (1920×1080) in (a) to (c) are referred as HTI-1 to HTI-3, respectively.	185

List of Tables

Table 2.1	Conditions used in classification of DCT blocks.	36
Table 3.1	Required number of multiplications for element-by-element multiplication of twiddle factor matrices and 2-D subband transform, where R and C denote the same element is found in the row and column of $\mathbf{W}_{cc}./\mathbf{B}$, $\mathbf{W}'_{cs}./\mathbf{B}$, $\mathbf{W}'_{sc}./\mathbf{B}$, and $\mathbf{W}'_{ss}./\mathbf{B}$, respectively.	54
Table 3.2	Derivation of computational cost of proposed FSA.	67
Table 3.3	Computational cost of approximation of DCT-II using FSA.	68
Table 3.4	Normalized aliased components of $\mathbf{C}_{\mathbf{n}}^{\text{ApD1}}$.	73
Table 3.5	Normalized aliased component at $C(\mathbf{k}', \mathbf{n})$ of $\mathbf{C}_{\mathbf{n}}^{\text{ApD1}}$.	74
Table 3.6	Alias contribution of approximation of DCT-II.	76
Table 3.7	Normalized aliased component $C(k'_1, k_2, \mathbf{n})$ for $N = 4$.	78
Table 3.8	Normalized aliased component $C(k_1, k'_2, \mathbf{n})$ for $N = 4$.	78
Table 3.9	Normalized aliased component $C(\mathbf{k}', \mathbf{n})$ for $N = 4$.	79
Table 3.10	PSNR and MSSIM values of images reconstructed using AdS1.	82
Table 3.11	PSNR and MSSIM values of images reconstructed using AdS2.	86
Table 4.1	Parameters of various MTFs.	107
Table 4.2	Description of subjective ratings used in subjective evaluation.	116
Table 4.3	Subjective rating for noise-contaminated images (PSNR = 30 dB) from various JND models.	123
Table 4.4	Subjective rating for compressed images obtained with proposed JND model using various values of β in error pooling.	130
Table 4.5	Performance measures of proposed JND using various β used error pooling.	134

Table 4.6	Subjective rating for compressed images from various PICs.	136
Table 4.7	Regions of compressed images using Watson's PIC having sharp edges.	137
Table 4.8	Regions of compressed images using Wei's PIC having blur edges.	137
Table 4.9	Performance measures of various PICs.	140
Table 4.10	Average computational cost of PSBDCT.	141
Table 5.1	Determining sign of entries of $\mathbf{A}_{(Q,N)}$.	151
Table 5.2	Computational cost of proposed method.	158
Table 5.3	Downsizing by factor of P/Q using various block size $\tilde{N}$.	161
Table 5.4	PSNR values for downsizing by factor of $2/3$ and $3/4$ with various block sizes.	166
Table 5.5	MSSIM values for downsizing by factor of $2/3$ and $3/4$ with various block sizes.	167
Table 5.6	PSNR and perceptual distortion score of images downsized by a factor of $2/3$.	176
Table 5.7	PSNR and perceptual distortion score of images downsized by a factor of $3/4$.	177
Table 5.8	Mean PSNR and perceptual distortion score for downsizing of images by a factor of P/Q .	177
Table 5.9	Downsizing of images by a factor of P/Q using various methods.	181
Table 5.10	PSNR values for downsizing by a factor of $2/3$ using various methods.	182
Table 5.11	PSNR values for downsizing by a factor of $3/4$ using various methods.	183
Table 5.12	Average PSNR values for downsizing images using various methods.	184
Table 5.13	PSNR values for downsizing by a factor of $P/Q \times R/S$.	185
Table 5.14	Computational cost for downsizing by a factor of $P/Q \times R/S$.	187

List of Abbreviations and Acronyms

1-D	One-Dimensional
2-D	Two-Dimensional
ApD	Approximation Of Type-II DCT
AdS	Adaptive Subband Discrete Cosine Transform
ADM	Average Differential MSSIM
ADP	Average Differential PSNR
Bpp	Bits Per Pixel
cpd	Cycles Per Degree
CSF	Contrast Sensitivity Function
DMSSIM	Differential Mean Structural Similarity
DPSNR	Differential Peak Signal-To-Noise Ratio
DCT	Discrete Cosine Transform
DCT-II	Type-II DCT
DFT	Discrete Fourier Transform
DOM	Difference Of The Mesa
DST	Discrete Sine Transform
DSP	Digital Signal Processor
DWT	Discrete Wavelet Transform
FSA	Fast SBDCT Algorithm
GQMF	Generalized Quadrature Mirror Filterbank
HD	High-Definition
HF	High-Frequency
HVS	Human Visual System
IDCT-II	Inverse Type-II Discrete Cosine Transform
JND	Just-Noticeable-Difference
KLT	Karhunen-Loève Transform

LGN	Lateral Geniculate Nucleus
LF	Low-Frequency
MOS	Mean Opinion Scores
MSA	Mean Square Alias
MSE	Mean Square Error
MSSIM	Mean Structural Similarity
MF	Mid-Frequency
MTF	Modulation Transfer Function
MAC	Multiply And Accumulate
NAMM	Non-Linear Additivity Model for Masking
PSBDCT	Perceptual Subband Discrete Cosine Transform
PSNR	Peak Signal-To-Noise Ratio
PIC	Perceptual Image Coder
PSF	Point Spread Function
RHS	Right Hand Side
RMS	Root Mean Square
SBDCT	Subband Discrete Cosine Transform
SSIM	Structural Similarity
QMF	Quadrature Mirror Filterbank
VDM	Visual Discrimination Model
VDP	Visual Differences Predictor
VQEG	Video Quality Experts Group

List of Symbols

$\text{ADP}(\tilde{N})$	Average differential PSNR for block size $\tilde{N}$
$\text{ADM}(\tilde{N})$	Average differential MSSIM for block size $\tilde{N}$
$\mathbf{A}_{(Q,N)}$	Spatial relationship of QN -point DCT block and Q neighboring N -point DCT block
$bt(\mathbf{n})$	Block classification result of $\mathbf{n}$ th sub-image
(c_m, c_a)	Computational cost where c_m and c_a represents the number of multiplications and additions
$C(b, k_1, n_1)$	Subband transforms of 1-D SBDCT
$c(p_1, n_1)$	1-D input sequence
$c(\mathbf{p}, \mathbf{n})$	2-D input sequence or sub-image
$c(b_i, \mathbf{q}, \mathbf{n})$	Subbands of $N \times N$ input sequence or sub-image
$C(b_i, \mathbf{k}, \mathbf{n})$	Subband transforms of 2-D SBDCT
$C(\mathbf{k}, \mathbf{n})$	$\mathbf{k}$ th DCT coefficient at $\mathbf{n}$ th block
$C(k_1, n_1)$	k_1 th DCT coefficient at n_1 th block
$\mathbf{C}_{b_i, n_1}^{1D}$	Matrix representation of $C(b_i, k_1, n_1)$
$\mathbf{C}_{b_i, \mathbf{n}}^{2D}$	Matrix representation of $C(b_i, \mathbf{k}, \mathbf{n})$
$\mathbf{C}_{\mathbf{n}}$	Matrix representation of DCT coefficients at $\mathbf{n}$ th block
$\mathbf{c}_{\mathbf{n}}$	Matrix representation of $\mathbf{n}$ th sub-image of an image
$\mathbf{c}_{b_i, \mathbf{n}}$	Matrix representation of $c(b_i, \mathbf{q}, \mathbf{n})$
$\hat{C}(\mathbf{k}, \mathbf{n})$	quantized $N \times N$ DCT or SBDCT coefficients of the $\mathbf{n}$ th sub-image
$e_1(\mathbf{n})$	Elevation parameter for luminance adaptation
$e_2(\mathbf{k}, \mathbf{n})$	Elevation parameter for contrast masking
$em(\mathbf{p}, \mathbf{n})$	Binary edge map of the $\mathbf{n}$ th sub-image
$f(\mathbf{x})$	Spatial frequency at $\mathbf{x}$
$f_{\text{DCT}}(\mathbf{k})$	Spatial frequency with respect to $\mathbf{k}$ th DCT subband
$\mathbf{H}$	Decimation filter matrix
H	Height of an image

$\mathbf{J}$	Downsizing filter
$\mathbf{J}^+$	Optimal least-square upsizing filter
$\mathbf{k}$	$\mathbf{k}$ th DCT subband, where $\mathbf{k} = (k_1, k_2)$
$MTF(\mathbf{x})$	MTF where spatial frequency is determined by $\mathbf{x}$
N	Length of input sequence of DCT
$T(\mathbf{k}, \mathbf{n})$	JND threshold of $\mathbf{k}$ th DCT subband of $\mathbf{n}$ th sub-image
$T_D(\mathbf{k}, \mathbf{n})$	Base detection threshold from spatial CSF
$T_b(\mathbf{k}, \mathbf{n})$	Base detection threshold of $\mathbf{k}$ th DCT subband from spatial CSF
$t'(\mathbf{x})$	JND threshold of a pixel at $\mathbf{x}$, obtained from $T(\mathbf{k}, \mathbf{n})$
$t(\mathbf{x})$	JND profile at $\mathbf{x}$
$t_B(b_j, \mathbf{s})$	b_j th subband JND profile at $\mathbf{s}$, where $\mathbf{s} = (s_1, s_2)$
$t_{\text{NAMM}}(\mathbf{x})$	JND threshold of a pixel at $\mathbf{x}$, obtained from a non-linear model for masking
$T_{la}(\mathbf{x})$	Luminance adaptation at $\mathbf{x}$
$T_{tm}(\mathbf{x})$	Texture masking at $\mathbf{x}$
t	Transpose operator
$\mathbf{T}_c$	DCT basis function
$\mathbf{T}_s$	DST basis function
$T_{b, \text{MTF}}(\mathbf{k})$	Base visibility threshold of $\mathbf{k}$ th DCT subband from MTF
V	Viewing distance
W	Width of an image
$W_c^{a/N}$	Cosine function defined as $\cos(a\pi/N)$
$W_s^{a/N}$	Sine function defined as $\sin(a\pi/N)$
$\mathbf{x}$	Location of pixel of image, where $\mathbf{x} = (x_1, x_2)$
ρ_p	Pearson correlation
ρ_s	Spearman rank-order correlation

Chapter 1

Introduction

In this new millennium, we have witnessed a tremendous growth in the use of digital images as an effective mean to represent and communicate information. Coupled with the advances in digital imaging technology, computing capability of processor, storage capacity and telecommunication, digital images have become an integral part of our daily lives.

One of the biggest challenges in the field of image and video processing is evaluating and optimizing the quality of digital imaging system with respect to the available storage and transmission bandwidth. The human visual system (HVS) is unable to detect all changes in an image due to its physiological and psychological mechanisms. By exploiting the limitations of the HVS, scarce resources of digital imaging system (such as storage space and transmission bandwidth) can be optimally allocated for the optimal visual experience.

With the emergence of mobile multimedia capable devices and availability of broadband networks, users have easy access to high definition image and video contents. The mobile devices available to users usually have smaller displays as well as limited processing capabilities. Interoperability between different screen sizes becomes important, hence, an efficient method to reduce the spatial resolution of image or video content before delivery to these mobile devices is needed [VCS03].

Discrete cosine transform (DCT) [ANR74] is used in many image and video standards (JPEG, MPEG-1/2/4, H.261/3), therefore, this thesis focuses on

computational model for just-noticeable-difference (JND) that are based on DCT subbands and pixels. These computational models have heavily influenced the design of many image quality metrics and perceptual image coders (PICs). Specifically, computational models for JND have been incorporated into many PICs [SJ89, Wat93, CC95, CC96, TS96, CB99, HK00, HK02, YLL03, YLL05, ZLX05, ZLX08, WN09, LLP10] to achieve high compression ratio without introducing perceptible distortion.

This thesis discusses the integration of computational models for JND and subband discrete cosine transform (SBDCT) [JMM96], under the JPEG framework. From the understanding of vision models of HVS, computational models for JND are derived. In particular, this thesis focuses on the computational models for JND based on DCT and pixel domains. Also, the computational structure for the SBDCT is studied, which led to the derivation of a fast computational structure for SBDCT. The application of computational models for JND is not limited to image compression, and can be adopted in other image processing techniques such as image resizing. On this note, the last part of this thesis discusses the derivation and analysis of a computationally efficient method to resize images in the DCT domain. Several key advantages of resizing images in the DCT domain over resizing images in the pixel domain are low computational cost as well as good image quality [DA01, PPO03, ST04, MM05, PP06, HC07, LL07]. However, the computational cost to resize images by some fractional factors can still be prohibitive. By incorporating computational models for JND into the image resizing method, the problem of high computational cost can be mitigated without impacting the image quality of resized images.

This chapter begins with a brief description of the anatomy of the HVS. This description shall emphasize on the features of the HVS that have greatly influenced the implementation of many image quality metrics, computational models for JND, and PICs. A review of several image quality metrics, computational models for JND and PICs shall be discussed in the following chapter, followed by the derivation and analysis of a fast computational structure for SBDCT in Chapter 3. Chapters 4 and 5 discuss about image compression and image resizing, as well as their integration with computational models for JND. Sections 1.2 and 1.3 detail the motivation and contribution of this thesis, respectively. Finally, an overview of the chapters of this thesis is presented in Section 1.4.

1.1 *Anatomy of the Early Human Visual System*

Through an understanding of the fundamentals of the HVS, the design principle of computational models for JND and image quality metric can be determined. Also, this understanding can be applied into other image processing techniques such as image enhancement [LDX03] and watermarking [WPD99].

Figure 1.1 illustrates the early HVS [BGS82], which can be divided into four stages, namely, optical processing, retinal processing, lateral geniculate nucleus (LGN), and cortical processing. Light from our surrounding first strikes the eye at the cornea, and enters the eye through the pupil. Light then passes through the lens of the eye. Together with the cornea, the lens focuses an image onto the back of the eye where the retina resides. The retina contains light sensitive photoreceptors, which converts light energy into electro-chemical signals to be used in the nervous

Fig. 1.1 Four main processing blocks of the early HVS consisting of the optics of the eye, retina, LGN, and visual cortex. Adapted from [ZB06].

system. Subsequently, these electro-chemical signals are transmitted out of the eye via the optic nerves for further processing in the HVS.

Figure 1.2 illustrates the optics of the eye, which consists of the cornea, the pupil, and the lens. The relation between the visual angle, viewing distance, and height of an object is depicted in Fig. 1.3. The visual angle θ of the eye is a function of the viewing distance d (from the lens of the eye to the object viewed by the eye), and the height h of object viewed by the eye. Using simple trigonometry, the visual angle is determined as

$$\theta = 2 \tan^{-1} \left(\frac{h}{2d} \right) \text{ radian.} \quad (1.1)$$

The overall processing of the optics in the eye is approximately characterized by a lowpass filter. Hence, the image casted on the retina (commonly known as the retina image) can be obtained by convolving the input image with a point spread

Fig. 1.2 Optics of the eye.

Fig. 1.3 Visual angle of the eye.

function (PSF), such as the one proposed by Westheimer [Wes86], and is expressed as

$$\text{PSF}(r) = 0.952e^{-2.59|r|^{1.36}} + 0.048e^{-2.43|r|^{1.74}}, \quad (1.2)$$

where r is the radial distance in minutes of arc from the center of the image that is aligned to the visual axis. The PSF function is plotted in Fig. 1.4.

The retina consists of two types of light sensitive photoreceptors, namely, rods and cones. The rods support achromatic vision in low light conditions, and the cones are responsible for daylight vision. There are three types of cones known as the short wavelength sensitive cones (440 nm), medium wavelength sensitive cones (540 nm), and long wavelength sensitive cones (570 nm). All the cones do

Fig. 1.4 Point spread function of the human eye as a function of distance [Wes86].

not encode color information, but color vision arises from the differential spectral sensitivities of the three cones. The signals from the photoreceptors pass through several layers of neurons (neurons are nerve cells, and in this case, bipolar cells, amacrine cells, and horizontal cells) before exiting the eye via the optic nerve (axons of the ganglion cells).

From Fig. 1.5, it is clear that the spatial distribution of the cone and rod photoreceptors is highly non-uniform. The highest density of cones is found at the fovea (0 degree of Fig. 1.5). The fovea region samples the retina image with the highest density, and permits the eye to perceive the highest resolution within this region. Due to the rapid drop of cones outside the fovea region, this resolution reduces rapidly with increasing distance from the fovea region. Since there is no photoreceptor at the optic nerve, this region becomes a blind spot of the eye.

After the information is encoded by the retina, it is transmitted to the LGN. Subsequently, the processed information is transmitted to the visual cortex. The

Fig. 1.5 Density of cones and rods as a function of eccentricity. Adapted from [Ost35].

visual cortex is made up of several layers, which are known as V1 (primary visual cortex or striate cortex), V2 (prestriae cortex), V3, V4, V5 (middle temporal), and V6. The primary visual cortex is directly connected to the LGN. One of the striking features of the neurons found in the primary visual cortex is the neurons are tuned to visual stimuli with specific spatial location, frequency, and orientation [Dau80, Mar80, WD85].

Based on the complexity of receptive field of the neurons in primary visual cortex, the neurons are classified as simple and complex. It has been suggested that the complex cell can be formed by summing simple cells [Dat82, Hee91, Cor05]. In addition, the receptive field of a generic simple cell can be modelled using a Gabor function [Dau80, Mar80, WD85]. Watson proposed the cortex transform [Wat87] based on the receptive field of a simple cell, and Watson's cortex transform was later adapted by Daly as the cortex filters of visible difference predictor (VDP) [Dal92, Dal93].

1.2 Motivation of Thesis

The ever growing need for storing and transmitting imagery content has driven the development of many image compression techniques. Currently, many of the image and video compression (JPEG, MPEG-1/2/4, H.261/3) are based on DCT. While there is rapid advancement in the research of discrete wavelet transform (DWT), it is interesting to note that the latest digital video storage media, such as Blu-ray still employs video compression techniques that are based on DCT.

With the advent of multimedia-capable mobile devices, interoperability between different systems (having different screen resolutions and screen sizes), and different broadband networks become important. To facilitate a seamless exchange of high-definition (HD) imagery content between different systems, an efficient technique to reduce the spatial resolution of such content before transmission to mobile devices is highly desirable. With most of the multimedia content stored and transmitted in compressed formats based on the DCT, many image transcoding techniques to resize images in the DCT domain are proposed [DA01, PPO03, SL04, MM05, PP06, LL07, SC07].

From the high adoption rate of DCT-based coders, it is obvious that techniques, which analyze and enhance the performance of DCT-based coders are extremely useful. Since the end-receiver of most of the image and video contents is human, it is vital that the limitations of the HVS are incorporated into such coders. On this note, the HVS is only capable of discerning differences in an image when the difference is above its visibility threshold, commonly referred as the JND threshold. This JND threshold is dependent on the viewing conditions, such as viewing distance and ambience lighting.

Usually, the original image is decomposed by a filterbank such as the quadrature mirror filterbank (QMF) used by the coders proposed in [SJ89, CL95, CC96, HK00]. However, such approaches are not compatible with DCT-based coders. On the other hand, several PICs based on DCT were proposed in [Wat93, ZLX05, ZLX08], which are compatible with JPEG [ISO94]. These PICs are successful in incorporating the visibility threshold into the quantization matrix. Due to the local varying content in images, local adaptation of the quantization matrix is often required. Since the same quantization matrix is required at the encoder and decoder to recover the compressed image, the overall bit-rate of compressed image increases (i.e. bits required for transmitting the quantization matrix). To overcome this problem, Watson [Wat93] fixed the adaptation of the quantization matrix to minimize the overall perceptual distortion of the image. On the other hand, Höntsch and Karam [HK00, HK02] eliminated the need to transmit the quantization matrix by estimating the quantization matrix from the quantized DCT coefficients.

The goal of this thesis is to derive a set of JND threshold based on the SBDCT, and applying this set of threshold for different viewing conditions, such as screen resolution and screen size. Subsequently, this set of visibility threshold is applied in two image processing algorithms, namely, image compression and image resizing.

1.3 Contribution of Thesis

The major contributions of this thesis can be summarized as follows.

- (i) Based on a symmetry property found in the SBDCT coefficients, a fast SBDCT algorithm (FSA) based on the $N/2$ fast 1-D DCT and $N/2 \times N/2$ fast 2-D DCT algorithms is derived.
- (ii) Due to the subband decomposition of the input sequence in SBDCT, the computational structure for SBDCT is well suited for parallel processing. By studying the computational structure for SBDCT, seven algorithms to approximate the Type-2 DCT (DCT-II) is derived. The mean square error (MSE) between the approximation algorithms and DCT-II as well as the aliased components of the seven approximation algorithms are analyzed.
- (iii) A computational model for JND based on the SBDCT is formulated. This led to a unique integration of the JND model and SBDCT, which takes into account of the computational structure for the SBDCT. Several important modules of the proposed PIC such as edge detection as well as block classification are specifically designed with the computational structure for the SBDCT in mind. Furthermore, the computational models for JND in the pixel and DCT domains are applied in various stages of the proposed PIC to achieve superior image quality as compared to several existing PICs. The superior performance of the proposed PIC is backed up by subjective evaluation, and it is also shown that the amount of distortion predicted by the proposed PIC correlates well with subjective evaluation.
- (iv) Using the observation of the SBDCT as well as a spatial relationship of DCT-II coefficients between a block and its sub-blocks, an efficient transcoding method to arbitrarily resize images in the DCT domain is derived. This technique is designed with $N \times N$ DCT-II blocks as the input

to facilitate easy integration of the proposed method with existing image and video coders.

- (v) A perceptually-tuned image transcoding method is formulated by incorporating JND threshold. By using the JND threshold to selectively process the perceptually important DCT-II coefficients only, significant reduction of the computational cost associated with image resizing is achieved at no expense of visual quality of the resized images.

1.4 Thesis Overview

This thesis is organized into six chapters.

Chapter 1 serves as the introduction to the topic of the thesis, which involves the HVS. The fundamental concepts of the HVS are reviewed along with the anatomy of the human eye. This chapter also gives the motivation for incorporating the properties of the HVS into image compression and image resizing.

Chapter 2 reviews the fundamental concepts of the HVS, and presents the derivation of a set of visibility threshold. The theory and formulation of the important properties of HVS are studied and summarized. This chapter focuses on the JND threshold based on the DCT subbands, and a discussion on the JND threshold estimated from pixels is also presented. Several useful techniques involving the transformation of the JND threshold of pixels to the visibility threshold of DCT subbands and vice versa are discussed.

Chapter 3 begins with the review of the SBDCT, and then presents the derivation of the FSA. The main difference between the DCT-II and the SBDCT lies in the subband decomposition that is performed on the input sequence in the

case of SBDCT. This decomposition results in more coefficients, and higher computational cost as compared to the DCT-II. However, image decomposition generally leads to a parallel structure in implementation. Based on the symmetric property found in the SBDCT coefficients, an $N \times N$ FSA based on an $N/2 \times N/2$ fast 2-D DCT-II algorithm is proposed. To achieve further computational cost reduction, seven approximations of the DCT-II based on SBDCT are derived by using different combinations of the subband transforms of the SBDCT. This derivation is accompanied with the analysis on the MSE and the aliased components of these seven approximation algorithms.

Chapter 4 investigates the integration of a computational model for JND and SBDCT, which leads to the derivation of the proposed PIC. The computational model for JND is first estimated from pixels of an image, and then converted to the visibility threshold based on DCT subbands. Specifically, the computational model for JND is used to select the DCT-II approximation algorithms derived in Chapter 3, and to determine the quantization matrix that is adapted to the local content of an image. In other words, both the computational models for JND in pixel and DCT domains are adopted at various stages of the proposed PIC. Block classification is adopted in many recent PICs [ZLX05, ZLX08, WN09], and are found to greatly improve the performance of many PICs. To reduce computational cost, the intermediate results from SBDCT are reused for edge detection, and subsequently block classification of the image. The second part of this chapter focuses on a comparative study of various PICs. This study is carried out using subjective evaluation to compare various PICs in terms of image quality at a given bit-rate. Also, the correlation of the predicted distortion from each PIC between the subjective rating is analyzed.

The first part of Chapter 5 deals with image transcoding to arbitrarily resize images in the DCT domain using observations of the SBDCT, and the spatial relationship of DCT-II coefficients of a block and its sub-blocks. The performance of this algorithm is characterized in terms of computational cost, image quality, and the aliasing introduced by the resizing filters. To facilitate easy integration of this algorithm with image and video coders, this algorithm is formulated using only $N \times N$ DCT-II blocks as input. The first derivation of this algorithm leads to resized images with good image quality, but requiring high computational cost. Subsequently, we show that it is possible to significantly reduce the computational cost of this algorithm by using a subset of the up-sampled coefficients in the down-sampling stage with minimum degradation on image quality. The second part of Chapter 5 details an extension of the image transcoding method using the properties of the HVS that are formulated in Chapter 4. This permits the selection of resizing filters based on the local image content. Not only the target display resolution is considered, this method also takes into account the height and width of each pixel on the display.

Chapter 6 concludes this thesis with a summary of the major issues. Directions for future research are also discussed.

Appendix A contains the derivation for the aliased components of the seven approximation algorithm presented in Chapter 3. Appendix B contains the test-images used in the simulations for this thesis. Appendix C presents the objective measurements of the resized images obtained from the simulations conducted in Chapter 5.

Chapter 2

Perceptual Image Coding Based on Human Vision System

The growing demand for transmission and storage of images has spurred much effort in improving image compression techniques. To achieve this goal, one promising approach is to integrate properties of the HVS into image compression techniques [JJS93]. The central idea of such approach is to embed coding distortion beneath the spatial visibility threshold of the HVS. This threshold is commonly referred as the JND threshold [JJS93] as it specifies the minimum sensory difference that is detectable by the HVS. In the context of image compression, if the coding error of each pixel in a compressed image is exactly at level of JND, a perceptually perfect image will be obtained at the lowest possible bit-rate [JJS93]. Over the years, several computational models for JND have been developed and employed in image compression. These computational models for JND models are computed based on subbands [SJ89, Wat93, TS96, HK00, HK02, ZLX05, ZLX08, WN09] and pixels [CL95, CC96, CB99, YLL03, YLL05, LLP10] of an image.

The first few models of the HVS [Sch56, MS74, Fau79, LB82] were formed using a single channel approach. Such models regard the HVS as a single spatial filter, which is defined by the contrast sensitivity function (CSF). One of the first few HVS based image quality metrics for luminance images was developed by Mannos and Sakrison [MS74]. By inferring some properties of the human vision

from psychophysical experiments, Mannos and Sakrison derived a closed-form expression describing the contrast sensitivity of the HVS as a function of spatial frequency.

Due to the fact that the HVS is a multiple channels system with each channel tuned to different ranges of spatial frequencies and orientations [Dau80], many multi-channel models were subsequently proposed. Multi-channel HVS models are employed in metrics such as VDP proposed by Daly [Dal92, Dal93], and the visual discrimination model (VDM) proposed by Lubin [Lub93, Lub95]. These image quality metrics are intended for general applicability, and are computationally expensive to implement.

On the other hand, a priori knowledge of the image processing algorithm (such as image compression) permits the use of specialized vision models. Specialized vision models are not as versatile as generalized models, but these specialized models can perform very well in a given application scope. Such vision models are usually simpler and computationally efficient. One example of an image coder based on a specialized vision model is the DCTune [Wat93]. By exploiting the limitations of the HVS, DCTune permits higher compression ratio while preserving the visual quality of the compressed image.

In general, most vision models incorporate four properties of the HVS, namely, spatial contrast sensitivity, luminance adaptation, contrast masking, and temporal masking [Bov05]. The spatial contrast sensitivity of the HVS is parameterized by the CSF, which is typically of bandpass response. A set of contrast sensitivity is obtained from the CSF, which states the amount of contrast at a spatial frequency to be detectable in a flat gray image. This contrast sensitivity increases with background luminance as stated by Weber's Law. This

is true for a wide range of luminance except for very low or high light conditions. The effect of the background luminance on contrast sensitivity is usually accounted as luminance adaptation, or luminance masking [Wat93]. Contrast masking refers to the change of the visibility of one image component due to the presence of another. The strongest contrast masking occurs when both components are of the same or at similar spatial frequency, orientation, and location. Temporal variation of light intensity falling into the eye can also reduce contrast sensitivity. This reduction of the contrast sensitivity is referred as the temporal masking, and is commonly adopted in video compression. Since this thesis focuses on image processing, the first three properties of the HVS will be introduced in the following sections of this chapter.

The three properties of the HVS can be integrated into a PIC as shown in Fig. 2.1. Considering the fact that the HVS is a multi-channel system, the frequency decomposition block decomposes the input image into several components (also known as channels or subbands). Numerous decomposition methods are used in PICs and image quality metrics, which include Fourier decomposition [CR68, MS74], Gabor decomposition [Dau88, LB90], DCT [Wat93, HK02, YLL03, YLL05, ZLX05, ZLX08], wavelet transform [TH94a, WHM97, LK00], and polar separable wavelet transform [Wat87, TH94b]. Please refer to the survey paper [LK11] for discussions on many recent image quality metrics. In addition, many image quality metrics and PICs implement error pooling after CSF, luminance adaptation, and contrast masking. The role of error pooling [Wat79, RB81] is to combine the errors that have been computed for each spatial frequency, orientation band, and location into a single number or a distortion map.

Fig. 2.1 Block diagram of PIC.

As DCT is widely used in image and video compression (e.g. JPEG, MPEG-1/2/4, H.261/3), we focus our discussion in this chapter on image quality metrics computed using DCT subbands. In addition, it is useful to consider the pixel-based image quality metrics since it is possible to convert the contrast sensitivity from the pixel domain to the subband domain and vice versa.

The objective of this chapter is to review the concepts from the psychophysics of human vision that are applied to image quality metrics and computational models for JND. In particular, this chapter emphasizes on image quality metrics and computational models for JND that are designed specifically for image compression. The frequency decomposition using cortex filters is first introduced in Section 2.1. This is followed by mapping from cortex filters to DCT-II subbands (or coefficients). In Section 2.2, the widely adopted spatial CSF proposed by Ahumada and Peterson [AP92] is discussed. Due to the fact that detection threshold is inversely proportional to contrast sensitivity, the detection threshold for every DCT subband is derived from a spatial CSF. It is well known that the detection threshold is varied by the local mean luminance and local spatial content. As mentioned earlier, these adjustments to the detection threshold are referred as luminance adaptation and contrast masking, respectively. Section 2.3 illustrates the effects of luminance adaptation using Weber's law, which states that the sensitivity of the HVS increases with the background luminance.

Subsequently, several techniques estimating the luminance adaptation from the subbands and pixels of an image are reviewed. Next, intra- and inter-band contrast masking are introduced in Section 2.4. Intra-band contrast masking is typically adopted in many PICs due to its simple formulation. A discussion on estimating inter-band masking using cortex filters and block classification is also included. The final step of many image quality metrics, known as error pooling, is reviewed in Section 2.5. Since the JND threshold of a spatial frequency can be considered as a weighted sum of a group of pixels, it is possible to obtain the JND threshold of pixels from subband JND threshold and vice versa. In Section 2.6, we shall discuss the techniques that convert JND threshold estimated from pixels to subbands or vice versa.

2.1 Frequency Decomposition

The multi-channel response of HVS approximates a dyadic system [Dau80] that is well-matched by a multi-resolution filterbank or a wavelet decomposition. Examples of multi-resolution filterbank are cortex transform [Wat87] and cortex filter [Dal92, Dal93]. The cortex transform was first conceived by Watson [Wat87], which originated from neurophysiology [HW62, DAT82] and psychophysical studies in masking [BC69, SJ72]. The cortex transform is then adapted by Daly as the cortex filters in VDP. The decomposition of the frequency plane adopted by Watson and Daly is shown in Fig. 2.2. The main difference between the Watson's and Daly's implementation of the cortex filtering is that Daly used six orientation bands [PDT77, DYH82], instead of four (in the case of Watson's cortex transform), to better approximate the orientation selectivity of the HVS. Several HVS models [WR84, Wat87, Dal92, Dal93] use six spatial

Fig. 2.2 Decomposition of the frequency plane corresponding to (a) Watson's cortex transform [Wat87], (b) Daly's cortex filters [Dal92, Dal93], and (c) DCT-II. The range of each axis is from $-f_s/2$ to $f_s/2$, where f_s is the sampling frequency.

channels, and it was found that six spatial channels show good agreement with psychophysical data [WLM90].

2.1.1 Cortex Filters

The cortex filters model the spatial (or radial) frequency selectivity and the orientational selectivity of the HVS. These filters are formed by cascading two classes of filters, which model the radial frequency bands and orientation bands of the HVS. The radial frequency filters are formed as differences of a series of two

Fig. 2.3 Frequency response of a mesa filter [Dal92, Dal93].

dimensional (2-D) low-pass mesa filters, which possess a flat passband, a Hanning window transition band, and a flat stopband as shown in Fig. 2.3. The mesa filter [Dal92] is completely described by its half-amplitude frequency $d_{1/2}$ and transition width tw . Let d denote the spatial frequency in cycles per degree (cpd).

The mesa filter $\text{mesa}(d)$ is expressed as

$$\text{mesa}(d) = \begin{cases} 1, & \text{for } d < d_{1/2} - \frac{tw}{2}, \\ \frac{1}{2} \left[1 + \cos \left(\frac{\pi (d - d_{1/2} - tw/2)}{tw} \right) \right], & \text{for } d_{1/2} - \frac{tw}{2} \leq d \leq d_{1/2} + \frac{tw}{2}, \\ 0, & \text{for } d > d_{1/2} + \frac{tw}{2}. \end{cases} \quad (2.1)$$

where $tw = 2d_{1/2}/3$. The radial frequency selectivity of the HVS is modelled by the difference of two mesa filters with different half amplitude frequencies. The difference of the mesa (DOM) filter $\text{dom}(b_d, d)$ is given by

$$\text{dom}(b_d, d) = \text{mesa}(d)|_{d_{1/2}=2^{-(b_d-1)}} - \text{mesa}(d)|_{d_{1/2}=2^{-b_d}}, \quad (2.2)$$

where $b_d = 0, 1, \dots, D - 1$, and D is the number of DOM filters. The choice of tw yields a set of cortex bands with approximately constant behaviour on a log frequency axis with a bandwidth of one octave [SJ72, MTT78, DAT82]. The orientation sensitivity of the HVS is modelled by a set of fan filters [Dal92], which is expressed as

$$\text{fan}(\theta, b_f) = \begin{cases} \frac{1}{2} \left[1 - \cos \left(\frac{\pi |\theta - \theta_{cr}(b_f)|}{\theta_{tw}} \right) \right], & |\theta - \theta_{cr}(b_f)| \leq \theta_{tw}, \\ 0, & \text{otherwise,} \end{cases} \quad (2.3)$$

where θ_{tw} is the angular transition width in degree, and $\theta_{cr}(b_f)$ is the orientation of the center angular frequency of the b_f th fan filter in degree, $b_f = 0, 1, \dots, F - 1$, and F is the number of fan filters. $\theta_{cr}(b_f)$ is given by

$$\theta_{cr}(b_f) = (b_f - 1)\theta_{tw} - 90^\circ, \quad (2.4)$$

where $\theta_{tw} = 180^\circ/F$. The cortex filter at the $\mathbf{b}$ th band $\text{cortex}(\mathbf{b}, d, \theta)$ is formed by the product of the DOM and fan filters as

$$\text{cortex}(\mathbf{b}, d, \theta) = \begin{cases} \text{dom}(d, b_d) \text{fan}(\theta, b_f), & \text{for } b_d = 1, \dots, D - 1; b_f = 0, 1, \dots, F - 1, \\ \text{base}(d), & \text{for } b_d = D, \end{cases} \quad (2.5)$$

where $\mathbf{b} = (b_d, b_f)$, and $\text{base}(d)$ is the cortex filter having the lowest spatial frequency without orientational selectivity. In [TS96], $\text{base}(d)$ is implemented with a truncated Gaussian function given as

$$\text{base}(d) = \begin{cases} e^{-\frac{d^2}{2\sigma^2}}, & \text{for } d < d_{1/2} + \frac{tw}{2}, \\ 0, & \text{for } d \geq d_{1/2} + \frac{tw}{2}, \end{cases} \quad (2.6)$$

where $\sigma = (2^{-D} + tw/2)/3$. Six spatial channels ($D = 6$) and six orientation bands ($F = 6$) are used in Daly's implementation of the cortex filters.

2.1.2 Mapping DCT Coefficients to Cortex Bands

Tran and Safranek [TS96] estimated inter-band contrast masking by grouping the DCT subbands within the same cortex band (estimation of inter-band contrast masking using cortex band will be discussed in Section 2.4.1). Since the cortex filters model the spatial frequency selectivity and orientation selectivity of the HVS, it would be useful to consider the mapping of DCT-II coefficients to the cortex bands.

The general idea behind mapping of DCT-II coefficients to cortex bands is to group the DCT-II coefficients that belong to the same cortex bands. An example is illustrated in Fig. 2.4. In order to map the partially covered DCT-II coefficients that fall within a cortex band, Tran and Safranek divide each DCT-II coefficients into $M \times M$ smaller blocks referred as sub-bins. Subsequently, these sub-bins are grouped into the corresponding $\mathbf{b}$ th band of the cortex filters.

Let $\mathbf{k}$ denote the DCT-II coefficient of an $N \times N$ DCT block, where $\mathbf{k} = (k_1, k_2)$, $k_1, k_2 = 0, 1, \dots, N - 1$. The overlapping area between $\mathbf{k}$ th DCT-II coefficient, and the corresponding band cortex band is computed as

$$\text{Overlap}(\mathbf{k}, \mathbf{b}) = \sum_{m_1=k_1M}^{(k_1+1)M} \sum_{m_2=k_2M}^{(k_2+1)M} \text{cortex}(\mathbf{b}, m_1, m_2), \quad (2.7)$$

Fig. 2.4 Mapping of DCT coefficients (thin line) to cortex bands (thick line). The shaded area denotes the DCT coefficients which fall within the same cortex band. Dashed lines denotes the sub-bins of each DCT coefficient.

which leads to T_{CF} $N \times N$ matrices, where T_{CF} is the number of cortex filters.

Each T_{CF} matrix contains the information of the overlapping area of the N^2 DCT-II coefficients. There are as many as $F(D-1)+1$ cortex filters, but this algorithm only maps 21 bands as only one quadrant of the cortex space is mapped to the DCT-II space. The remaining bands ($31 - 21 = 10$ bands considering Daly's implementation of the cortex filters) are assumed to have zero overlapping area.

2.2 Spatial Contrast Sensitivity Function

A number of well known factors that affect the visibility of difference in an image are spatial CSF, luminance adaptation, and contrast masking. Generally, the JND threshold is formulated as a product of the detection threshold $T_b(\mathbf{k}, \mathbf{n})$, and its elevation parameters given by luminance adaptation and contrast masking. Luminance adaption and contrast masking of an $N \times N$ sub-image in an image located at $\mathbf{n}$ are denoted by $e_1(\mathbf{n})$ and $e_2(\mathbf{k}, \mathbf{n})$, respectively, where $\mathbf{n} = (n_1, n_2)$;

$n_1 = 0, 1, \dots, H/N - 1$; $n_2 = 0, 1, \dots, W/N - 1$; W denotes the width of an image, and H denotes the height of an image. The DCT-II [ANR74] of an $N \times N$ sub-image, $c(\mathbf{p}, \mathbf{n})$, is given as

$$C(\mathbf{k}, \mathbf{n}) = \alpha_{k_1} \alpha_{k_2} \sum_{p_1=0}^{N-1} \sum_{p_2=0}^{N-1} c(\mathbf{p}, \mathbf{n}) \cos\left(\frac{(2p_1+1)k_1\pi}{2N}\right) \cos\left(\frac{(2p_2+1)k_2\pi}{2N}\right), \quad (2.8)$$

where

$$\alpha_u = \begin{cases} \sqrt{\frac{1}{N}}, & u = 0, \\ \sqrt{\frac{2}{N}}, & u = 1, 2, \dots, N-1, \end{cases} \quad (2.9)$$

and $\mathbf{p} = (p_1, p_2)$.

In this section, the widely adopted spatial CSF [Wat93, HK02, YLL03, YLL05, ZLX05, ZLX08], which was proposed by Ahumada and Peterson [AP92], is discussed. Ahumada and Peterson's formulation of the CSF is very useful as their CSF takes into account of display luminance levels, veiling luminance levels and spatial frequencies. In the case of DCT-II, the measured threshold is the smallest contrast that yields a visible signal (corresponds to a DCT subband) over a uniform intensity. Since no masking is present in this measurement, the measured threshold is also referred as the base detection threshold $T_D(\mathbf{k}, \mathbf{n})$, and defines the detection threshold of the HVS with respect to the spatial frequency and the orientation of the DCT subband. To account for luminance adaptation and contrast masking, the JND threshold $T(\mathbf{k}, \mathbf{n})$ based on DCT subband becomes

$$T(\mathbf{k}, \mathbf{n}) = T_D(\mathbf{k}, \mathbf{n}) e_1(\mathbf{n}) e_2(\mathbf{k}, \mathbf{n}). \quad (2.10)$$

Based on the van Nes and Bouman's measurements [NB67], Ahumada and Peterson approximated the detection threshold $T_D(\mathbf{k}, \mathbf{n})$ of the $\mathbf{k}$ th DCT subband

using a parabola in log spatial frequency. The detection threshold $T_D(\mathbf{k}, \mathbf{n})$ is expressed as

$$\log_{10}(T_D(\mathbf{k}, \mathbf{n})) = \log_{10}(T_{\min}(\mathbf{n})) + K(\mathbf{n})(\log_{10} f(\mathbf{k}) - \log_{10} f_{\min}(\mathbf{n}))^2, \quad (2.11)$$

$$k_1 = 0 \text{ or } k_2 = 0,$$

where $T_{\min}(\mathbf{n})$, $K(\mathbf{n})$, and $f_{\min}(\mathbf{n})$ are functions of the local luminance $L(\mathbf{n})$; $f(\mathbf{k})$ is the spatial frequency of a grating and is computed as $f(\mathbf{k}) = \sqrt{(k_1/w_x)^2 + (k_2/w_y)^2}$, where w_x and w_y are the horizontal width and vertical height of a pixel, respectively. The total luminance $L(\mathbf{n})$ is the sum of veiling luminance and the luminance of the image located at $\mathbf{n}$. $T_{\min}(\mathbf{n})$ is the luminance threshold at $f_{\min}(\mathbf{n})$, the frequency at which the threshold is the smallest, and $K(\mathbf{n})$ determines the steepness of the parabola.

Figure 2.5 summarizes the contrast sensitivity for luminance-only sine wave grating from [NB67]. This measurement data is taken for luminance at one decade interval from $2.9 \times 10^{-4} \text{ cd/m}^2$ to $2.9 \times 10^2 \text{ cd/m}^2$. As van Nes and Bouman found no difference between the CSFs for luminance between 290 cd/m^2 and 1880 cd/m^2 , Ahumada and Peterson used a maximum display luminance of 300 cd/m^2 in their experiments. At each luminance, a parabola was obtained for a range of spatial frequency up to 100 cpd. Generally speaking, the parabola moves up as luminance increases. It was reported in [Kel85] that the parabola model is not valid for low spatial frequencies, and Peterson *et al.* [PMP91] suggested a conservative estimate for $T_D(0,0,\mathbf{n})$, which is the smaller value of $T_D(1,0,\mathbf{n})$ and $T_D(0,1,\mathbf{n})$.

Fig. 2.5 Contrast sensitivity data from van Nes and Bouman [NB67].

Equation (2.11) describes the detection threshold for the case of a single grating. In the case of two gratings, there is usually an increase in the detection threshold as compared to the case in (2.11). For the detection threshold of two gratings, the first term on the right hand side (RHS) of (2.11) is changed to

$$\frac{T_{\min}(\mathbf{n})}{r + (1-r)\cos^2 \theta(\mathbf{k})}, \quad (2.12)$$

where $\theta(\mathbf{k}) = \sin^{-1}(2f(k_1, 0)f(0, k_2)/f^2(\mathbf{k}))$ and $r = 0.7$. Hence, the log of $T_D(\mathbf{k}, \mathbf{n})$ becomes

$$\log_{10}(T_D(\mathbf{k}, \mathbf{n})) = \log_{10}\left(\frac{T_{\min}(\mathbf{n})}{r + (1-r)\cos^2 \theta(\mathbf{k})}\right) + K(\mathbf{n})(\log_{10} f_{\mathbf{k}} - \log_{10} f_{\min}(\mathbf{n}))^2. \quad (2.13)$$

2.2.1 Determining Luminance Dependent Parameters of Spatial CSF

Using the experimental data in [NB67, PPMP91], the luminance-dependent parameters of the parabola describing the contrast sensitivity are found to be

$$T_{\min}(\mathbf{n}) = \begin{cases} \frac{L_T}{S_0} \left(\frac{L(\mathbf{n})}{L_T} \right)^{\alpha_T}, & L(\mathbf{n}) \leq L_T, \\ \frac{L(\mathbf{n})}{S_0}, & \text{otherwise,} \end{cases} \quad (2.14)$$

$$f_{\min}(\mathbf{n}) = \begin{cases} f_0 \left(\frac{L(\mathbf{n})}{L_f} \right)^{\alpha_f}, & L(\mathbf{n}) \leq L_f, \\ f_0, & \text{otherwise,} \end{cases} \quad (2.15)$$

and

$$K(\mathbf{n}) = \begin{cases} K_0 \left(\frac{L(\mathbf{n})}{L_K} \right)^{\alpha_K}, & L(\mathbf{n}) \leq L_K, \\ K_0, & \text{otherwise,} \end{cases} \quad (2.16)$$

where $\alpha_f = 0.182$, $\alpha_K = 0.0706$, $\alpha_T = 0.649$, $f_0 = 6.78$ cpd, $K_0 = 3.125$, $L_f = 300$ cd/m², $L_K = 300$ cd/m², $L_T = 13.45$ cd/m², and $S_0 = 94.7$. Since van Nes and Bouman found no difference between the CSF for luminance level of 290 cd/m² and 1880 cd/m², (2.14) to (2.16) are clipped at 300 cd/m².

Assuming an image is displayed on a gamma corrected screen, we can linearly map signal intensity values into luminance levels. The base detection threshold $T_b(\mathbf{k}, \mathbf{n})$ is obtained as

$$T_b(\mathbf{k}, \mathbf{n}) = \frac{MT_D(\mathbf{k}, \mathbf{n})}{\alpha_{k_1} \alpha_{k_2} (L_{\max} - L_{\min})}, \quad (2.17)$$

where M is the number of gray levels in the image, $L_{\max}$ and $L_{\min}$ are the maximum and minimum luminance levels of the display, respectively. Clearly, the

quantization of each DCT-II coefficient should not be greater than $2T_b(\mathbf{k}, \mathbf{n})$ to ensure the quantization error remains invisible to the HVS.

To estimate the local luminance of an image, we can make use of the DC DCT-II coefficient such as the approach defined in Watson's DCTune [Wat93]. In [HK02], Höntsch and Karam [HK02] estimated the local luminance from the foveal region. The highest acuity of the HVS is limited by the size of the foveal region, and typically covers two degrees of the visual angle. Based on the foveal region, Höntsch and Karam estimated the local luminance as

$$L(\mathbf{n}) = L_{\min} + \frac{L_{\max} - L_{\min}}{M} \left(\frac{\sum_{(0,0,m_1,m_2) \in F(0,0,\mathbf{n})} C(0,0,m_1,m_2)}{N_F N} \right), \quad (2.18)$$

where $F(0,0,\mathbf{n})$ denotes the foveal region centers at $\mathbf{n}$ in subband zero and covers two degrees of the visual angle; $C(0,0,m_1,m_2)$ denotes the DC DCT-II coefficient at (m_1, m_2) ; and N_F denotes the number of DCT-II coefficients at $\mathbf{n}$ in subband zero, which falls within the foveal region. N_F is computed as

$$N_F = \left(\left\lfloor \frac{2VR_x \tan(\theta/2)}{N} \right\rfloor \right) \left(\left\lfloor \frac{2VR_y \tan(\theta/2)}{N} \right\rfloor \right), \quad (2.19)$$

where V is the viewing distance in inches; R_x and R_y are the height and width of the display resolution in pixel per inch, and θ is the visual angle covered by the foveal region.

Since the luminance from a digital image spans a small luminance range in the spatial CSF experiment conducted by van Nes and Bouman [NB67], a single spatial CSF (based on the mean luminance of the image) maybe used for an image [ZLX05]. Hence, (2.13) can be simplified by replacing the total luminance $L(\mathbf{n})$

with the mean luminance L of the display [Wat92, ZLX05, ZLX08]. With this assumption, the detection threshold $T_D(\mathbf{k}, \mathbf{n})$ becomes $T_D(\mathbf{k})$, and the adaptation to the local luminance of an image is then adjusted using luminance adaptation [Wat93].

2.3 Luminance Adaptation

Weber's law is widely used to model luminance adaptation, and the Weber fraction

$$K = \frac{\Delta I}{I_{bg}}, \quad (2.20)$$

where I_{bg} is the background intensity, and ΔI is the just-noticeable incremental intensity over the background, is found to be nearly constant for a wide range of intensities [Hec24]. However, Weber's law does not hold for all range of background intensities and spatial frequencies. This can be shown by investigating the contrast sensitivity for sinusoidal gratings at various mean luminance levels. Figure 2.6 summarizes the contrast sensitivity for gratings modulated around mean luminance level of 0.1 cd/m^2 to 1000 cd/m^2 . At low spatial frequency, the contrast sensitivity remains relatively constant for mean luminance level of 10 cd/m^2 to 1000 cd/m^2 , which agrees well with Weber's law. However, this is not the case when the spatial frequency increases.

To account for luminance adaptation in the detection threshold, a common approach is to adjust the detection threshold using a function of the local luminance. On the other hand, Daly estimated the luminance adaptation by including a pixel-based nonlinear transformation before frequency decomposition

Fig. 2.6 CSFs at various mean luminance levels. Plots are calculated using the empirical CSF model in [Bar04]. The region where Weber's law holds is circled.

[SYD87]. Other commonly used transformations include power functions close to cube root [Sto72] and log functions [MS74].

Watson [Wat93] estimated luminance adaptation using

$$e_1^{\text{Wat}}(\mathbf{n}) = \left(\frac{C(0, 0, \mathbf{n})}{C_{L_{\text{mean}}}} \right)^{\alpha_T}, \quad (2.21)$$

where $C_{L_{\text{mean}}}$ refers to the DC DCT-II coefficient corresponding to mean luminance ($C_{L_{\text{mean}}} = 1024$ for a 8 bit image).

It is found in [SJ89, CL95] that higher visibility threshold occurs at very dark and bright regions of an image as compared to the regions having medium brightness. Hence, Zhang *et al.* [ZLX05, ZLX08] computed the luminance adaptation in the DCT domain as

$$e_1^{\text{ZLX}}(\mathbf{n}) = \begin{cases} 2 \left(1 - \frac{C(0,0,\mathbf{n})}{128N} \right)^3 + 1, & \text{for } C(0,0,\mathbf{n}) \leq 128N, \\ 0.8 \left(\frac{C(0,0,\mathbf{n})}{128N} - 1 \right)^2 + 1, & \text{otherwise.} \end{cases} \quad (2.22)$$

In the pixel domain, Chou and Li [CL95, YLL05] empirically determined the luminance adaptation of a pixel at $\mathbf{x}$, where $\mathbf{x} = (x_1, x_2)$, $x_1 = 0, 1, \dots, H-1$, and $x_2 = 0, 1, \dots, W-1$, using the following:

$$e_1^{\text{CL}}(\mathbf{x}) = \begin{cases} 17 \left(1 - \sqrt{\frac{L_s(\mathbf{x})}{127}} \right) + 3, & \text{for } L_s(\mathbf{x}) \leq 127, \\ \frac{3}{128} (L_s(\mathbf{x}) - 127) + 3, & \text{for } L_s(\mathbf{x}) > 127, \end{cases} \quad (2.23)$$

where $L_s(\mathbf{x})$ is the local luminance at $\mathbf{x}$, and (2.23) was obtained at a distance of six times of the image height. Chou and Li determined the local luminance $L_s(\mathbf{x})$ as

$$L_s(\mathbf{x}) = \frac{1}{32} \sum_{p_1=0}^4 \sum_{p_2=0}^4 i(x_1 - 2 + p_1, x_2 - 2 + p_2) B(p_1, p_2), \quad (2.24)$$

where $i(\mathbf{x})$ denote the pixel of an image at $\mathbf{x}$, and operator B is depicted in Fig. 2.7.

2.4 Contrast Masking

Contrast masking refers to the reduction of visibility of one image signal due to the presence of another signal. The masking characteristic of the HVS is known to be strongest when both signals are of the same spatial frequency, orientation, and location. Legge and Foley [LF80] conducted experiments using two gratings to determine the contrast masking effect, and their experimental results are shown in Fig. 2.8. Clearly, their results indicate that the masker can either hinder the

1	1	1	1	1
1	2	2	2	1
1	2	0	2	1
1	2	2	2	1
1	1	1	1	1

B

Fig. 2.7 Operator to determine average local luminance (B).

detection (masking) of the target or lead to a lower detection threshold (facilitation) of the target.

Contrast masking can be classified as inter- and intra-band masking. Sometimes, the term “texture masking” (inter-band masking) is used to refer to a “broadband” masker, where the masking effect is contributed by multiple frequency and orientation channels. On the other hand, intra-band masking refers to the masking due to the masker within the same frequency and orientation channel.

In [SJ89], Safranek and Johnston proposed a subband image coder employs a 4×4 band generalized QMF (GQMF) to decompose the image into 16 subbands. They defined inter-contrast masking as follows:

$$e_2^{\text{SJ}}(\mathbf{b}, \mathbf{n}) = \max \left(1, \left(\sum_{\mathbf{b}} w\text{CSF}(\mathbf{b}) \text{tex}^{\text{SJ}}(\mathbf{b}, \mathbf{n}) \right)^{0.15} \right), \quad (2.25)$$

where $\mathbf{b} = \{b_1, b_2\}$, $0 \leq b_1, b_2 \leq 3$, $\text{tex}^{\text{SJ}}(\mathbf{b}, \mathbf{n})$ is the texture energy of the $\mathbf{b}$ th subband at location $\mathbf{n}$, and $w\text{CSF}(\mathbf{b})$ is the $\mathbf{b}$ th weighting factor empirically derived from a CSF [Cor90]. The texture energy at location $\mathbf{n}$ of the $\mathbf{b}$ th subband is computed as

$$\text{tex}^{\text{SJ}}(\mathbf{b}, \mathbf{n}) = \begin{cases} \text{var}(\mathbf{n}), & \text{for } \mathbf{b} = (0, 0), \\ \text{energy}(\mathbf{b}, \mathbf{n}), & \text{otherwise,} \end{cases} \quad (2.26)$$

Fig. 2.8 Contrast threshold for detecting a 2 cpd grating (target) as a function of the contrast of a masking grating ranging from 1 to 4 cpd. Extracted from [LF80].

where $\text{var}(\mathbf{n})$ computes the variance at $\mathbf{n}$ of subband zero over the area given by (n_1, n_2) , $(n_1 + 1, n_2)$, $(n_1, n_2 + 1)$, and $(n_1 + 1, n_2 + 1)$, and $\text{energy}(\mathbf{b}, \mathbf{n})$ computes the energy at $\mathbf{n}$ of the $\mathbf{b}$ th subband.

Based on the masking model in [LF80], Watson [Wat93] adjusted the base detection threshold to account for intra-band masking using the following:

$$e_2^{\text{Wat}}(\mathbf{k}, \mathbf{n}) = \begin{cases} \max \left\{ 1, \left(\frac{|C(\mathbf{k}, \mathbf{n})|}{T_b(\mathbf{k}, \mathbf{n}) e_1^{\text{Wat}}(\mathbf{n})} \right)^{0.7} \right\}, & \mathbf{k} \neq (0, 0), \\ 1, & \mathbf{k} = (0, 0). \end{cases} \quad (2.27)$$

Since the effect of DC DCT-II coefficient on $t_b(\mathbf{k}, \mathbf{n})$ has been accounted in luminance adaptation, it is assumed that there is no contrast masking in the DC DCT-II coefficient. However, the DC DCT-II coefficient indirectly affects

contrast masking via $e_1^{\text{Wat}}(\mathbf{n})$ in the denominator of (2.27) for all DCT-II coefficients except for the DC coefficient.

Based on the estimation of contrast masking reported in [SJ89], Höntsch and Karam [HK00] proposed a more elaborate adjustment for contrast masking. To compute contrast masking, Höntsch and Karam estimated the intra- and inter-band masking. Let $ec_{\text{inter}}^{\text{HK}}(\mathbf{k}, \mathbf{n})$ and $ec_{\text{intra}}^{\text{HK}}(\mathbf{k}, \mathbf{n})$ denote the amount of intra- and inter-band masking at $\mathbf{n}$ of $\mathbf{k}$ th subband of an image, respectively. The elevation parameter $e_2^{\text{HK}}(\mathbf{k}, \mathbf{n})$ is computed as

$$e_2^{\text{HK}}(\mathbf{k}, \mathbf{n}) = ec_{\text{inter}}^{\text{HK}}(\mathbf{k}, \mathbf{n}) ec_{\text{intra}}^{\text{HK}}(\mathbf{k}, \mathbf{n}). \quad (2.28)$$

Höntsch and Karam estimated intra-band masking using a similar expression as (2.27), but with a lower exponent value of 0.36. Their adjustment for the intra-band masking is given as

$$ec_{\text{intra}}^{\text{HK}}(\mathbf{k}, \mathbf{n}) = \begin{cases} \max \left(1, \left(\frac{|C(\mathbf{k}, \mathbf{n})|}{T_b(\mathbf{k}, \mathbf{n})} \right)^{0.36} \right), & \mathbf{k} \neq (0, 0), \\ 1, & \mathbf{k} = (0, 0). \end{cases} \quad (2.29)$$

As for inter-band masking, Höntsch and Karam used (2.25) with an exponent of 0.035.

In another work performed by Höntsch and Karam [HK02], they estimated the local luminance, and computed the base threshold using (2.13) and (2.17). Based on the foveal region, they proposed the adjustment for intra-band masking as

$$e_2^{\text{HK2}}(\mathbf{k}, \mathbf{n}) = \begin{cases} \max \left(1, \left(\frac{|\bar{C}_F(\mathbf{k}, \mathbf{n})|}{T_b(\mathbf{k}, \mathbf{n})} \right)^{0.6} \right), & \mathbf{k} \neq \{0, 0\}, \\ 1, & \mathbf{k} = \{0, 0\}, \end{cases} \quad (2.30)$$

where $\bar{C}_F(\mathbf{k}, \mathbf{n})$ is the average magnitude of the DCT-II coefficients in the foveal region.

To estimate contrast masking, Zhang *et al.* [ZLX05, ZLX08] employed a block classification method [TV98], which was first proposed in [PJJ94]. To perform DCT block classification, the DCT-II coefficients of an $N \times N$ sub-image are divided into four sections as shown in Fig. 2.9. Let $L_T(\mathbf{n})$, $M_T(\mathbf{n})$, and $H_T(\mathbf{n})$ denote the sum of DCT-II coefficients (absolute magnitude) in the low-frequency (LF), mid-frequency (MF), and high-frequency (HF) sections, respectively, of the DCT-II block at $\mathbf{n}$. Let $tex_1^{ZLX}(\mathbf{n})$ denote the texture energy of an $N \times N$ sub-image at $\mathbf{n}$, and is computed as the sum of $M_T(\mathbf{n})$ and $H_T(\mathbf{n})$.

Fig. 2.9 DCT-II block classification for contrast masking. LF, MF, and HF are represented by the dark gray, light gray and white boxes, respectively.

The presence of an edge in the $N \times N$ sub-image at $\mathbf{n}$ is indicated by the measures $(L_T(\mathbf{n}) + M_T(\mathbf{n})) / H_T(\mathbf{n})$ and $L_T(\mathbf{n}) / M_T(\mathbf{n})$. A high ratio of $L_T(\mathbf{n}) / M_T(\mathbf{n})$ suggests that the edge is diagonally oriented [PJJ94, TV98]. In [ZLX05, ZLX08], the authors adapted these measures to $tex_2^{ZLX}(\mathbf{n}) = (\bar{L}_T(\mathbf{n}) + \bar{M}_T(\mathbf{n})) / \bar{H}_T(\mathbf{n})$ and $tex_3^{ZLX}(\mathbf{n}) = \bar{L}_T(\mathbf{n}) / \bar{M}_T(\mathbf{n})$, where $\bar{L}_T(\mathbf{n})$,

$\bar{M}_T(\mathbf{n})$, and $\bar{H}_T(\mathbf{n})$ are the means of $L_T(\mathbf{n})$, $M_T(\mathbf{n})$, and $H_T(\mathbf{n})$, respectively. Block classification is performed using $tex_1^{ZLX}(\mathbf{n})$, $tex_2^{ZLX}(\mathbf{n})$, and $tex_3^{ZLX}(\mathbf{n})$ as shown in Table 2.1. Each $N \times N$ sub-image is classified into one of the three classes, namely, PLAIN, EDGE, and TEXTURE. Consequently, sub-images that are generally smooth with few spatial activities are classified as PLAIN, sub-images containing a lot of complex spatial activities are classified as TEXTURE, and sub-images containing a clear edge are classified as EDGE.

Using the block classification result, inter-band contrast masking is computed by

$$ec_{inter}^{ZLX}(\mathbf{n}) = \begin{cases} 1 + \frac{tex_1^{ZLX}(\mathbf{n}) - 290}{1208}, & \text{for TEXTURE block,} \\ 1.25, & \text{for EDGE block \& } L(\mathbf{n}) + M(\mathbf{n}) > 400, \\ 1.125, & \text{for EDGE block \& } L(\mathbf{n}) + M(\mathbf{n}) \leq 400, \\ 1, & \text{for PLAIN block.} \end{cases} \quad (2.31)$$

Zhang *et al.* considered similar adjustment as (2.27) for intra-band contrast masking, and the amount of adjustment for contrast masking is obtained as

$$e_2^{ZLX}(\mathbf{k}, \mathbf{n}) = \begin{cases} ec_{inter}^{ZLX}(\mathbf{n}), & \text{for EDGE block} \\ & \text{at } \mathbf{k} \in LF \cup MF, \\ \max \left\{ ec_{inter}^{ZLX}(\mathbf{n}), \left(\frac{ec_{inter}^{ZLX}(\mathbf{n}) |C(\mathbf{k}, \mathbf{n})|}{T_b(\mathbf{k}, \mathbf{n}) e_1^{ZLX}(\mathbf{k}, \mathbf{n})} \right)^{0.36} \right\}, & \text{otherwise.} \end{cases} \quad (2.32)$$

To avoid over estimation of JND threshold at the EDGE blocks, the LF and MF regions of EDGE blocks are excluded from the estimation of intra-band contrast masking.

For a viewing distance of six times of the image height, Chou and Li [CL95] estimated contrast masking in the pixel domain using the following expression:

$$e_2^{CL}(\mathbf{x}) = 0.01L(\mathbf{x})(0.01G(\mathbf{x}) - 1) + 0.115G(\mathbf{x}) + 0.5, \quad (2.33)$$

Table 2.1 Conditions used in classification of DCT-II blocks.

Case	Condition	Block classification
I	$tex_1^{ZLX}(\mathbf{n}) \leq 125$	DCT block is classified as PLAIN
II	$125 < tex_1^{ZLX}(\mathbf{n}) \leq 290$ and $\max(tex_2^{ZLX}(\mathbf{n}), tex_3^{ZLX}(\mathbf{n})) \geq 7$ $\min(tex_2^{ZLX}(\mathbf{n}), tex_3^{ZLX}(\mathbf{n})) \geq 5$ or $tex_2^{ZLX}(\mathbf{n}) \geq 16$	DCT block is classified as EDGE, otherwise PLAIN.
III	$290 < tex_1^{ZLX}(\mathbf{n}) \leq 900$ and $\max(tex_2^{ZLX}(\mathbf{n}), tex_3^{ZLX}(\mathbf{n})) \geq 7$ $\min(tex_2^{ZLX}(\mathbf{n}), tex_3^{ZLX}(\mathbf{n})) \geq 5$ or $tex_2^{ZLX}(\mathbf{n}) \geq 16$	DCT block is classified as EDGE, otherwise TEXTURE.
IV	$tex_1^{ZLX}(\mathbf{n}) > 900$ and $\max(tex_2^{ZLX}(\mathbf{n}), tex_3^{ZLX}(\mathbf{n})) \geq 0.7$ $\min(tex_2^{ZLX}(\mathbf{n}), tex_3^{ZLX}(\mathbf{n})) \geq 0.5$ or $tex_2^{ZLX}(\mathbf{n}) \geq 16$	DCT block is classified as EDGE, otherwise TEXTURE.

where $G(\mathbf{x})$ is the maximal weighted average of the gradient around the pixel at $\mathbf{x}$. $G(\mathbf{x})$ is calculated by

$$G(\mathbf{x}) = \max_{j=1,2,3,4} \{ |grad_j(\mathbf{x})| \}, \quad (2.34)$$

where

$$grad_j(\mathbf{x}) = \frac{1}{16} \sum_{p_1=0}^4 \sum_{p_2=0}^4 c(x_1 - 2 + p_1, x_2 - 2 + p_2) G_j(p_1, p_2), \quad (2.35)$$

and $G_j(\mathbf{x})$ are four directional highpass filters as shown in Fig. 2.10.

2.4.1 Estimation of Inter-Band Contrast Masking from Cortex Bands

Tran and Safranek [TS96] estimated the amount of contrast masking using the texture energy found in each cortex band. This is motivated by the fact that

0	0	0	0	0
1	3	8	3	1
0	0	0	0	0
-1	-3	-8	-3	1
0	0	0	0	0
G_1				
0	0	1	0	0
0	8	3	0	0
1	3	0	-3	1
0	0	-3	-8	0
0	0	-1	0	0
G_2				
0	0	1	0	0
0	0	3	8	0
1	3	0	-3	1
0	-8	-3	0	0
0	0	-1	0	0
G_3				
0	1	0	-1	0
0	3	0	-3	0
1	8	0	-8	1
0	3	0	-3	0
0	1	0	-1	0
G_4				

Fig. 2.10 Operators to determine weighted average of luminance changes (G_p).

contrast masking is related to the texture energy in each cortex band. By using the mapping algorithm discussed in Section 2.1.2, the DCT-II coefficients are mapped to the cortex bands. The texture energy $tex^{TS}(\mathbf{b}, \mathbf{n})$ in each cortex band at $\mathbf{n}$ is calculated by

$$tex^{TS}(\mathbf{b}, \mathbf{n}) = \sqrt{\sum_{k_1=0}^{N-1} \sum_{k_2=0}^{N-1} \left(\frac{C(\mathbf{k}, \mathbf{n}) \text{Overlap}(\mathbf{k}, \mathbf{b})}{T_b(\mathbf{k}, \mathbf{n})} \right)^2}, \quad \text{for } \mathbf{k} \neq (0, 0). \quad (2.36)$$

By comparing $tex^{TS}(\mathbf{b}, \mathbf{n})$ with thresholds $\text{thres}_{\text{low}}$ and $\text{thres}_{\text{high}}$, Tran and Safranek computed the elevation factor for each cortex band as

$$e_2^{TS}(\mathbf{b}, \mathbf{n}) = \begin{cases} \min, & tex^{TS}(\mathbf{b}, \mathbf{n}) < \text{thres}_{\text{low}}, \\ \max, & tex^{TS}(\mathbf{b}, \mathbf{n}) > \text{thres}_{\text{high}}, \\ \min + \frac{\max - \min}{\text{thres}_{\text{high}} - \text{thres}_{\text{low}}} (tex^{TS}(\mathbf{b}, \mathbf{n}) - \text{thres}_{\text{high}}), & \text{otherwise,} \end{cases} \quad (2.37)$$

where $\max$, $\min$, $\text{thres}_{\text{low}}$ and $\text{thres}_{\text{high}}$ are determined using subjective evaluation.

2.5 Error Pooling

The final step of many image quality metrics is to combine the errors normalized by $T(\mathbf{k}, \mathbf{n})$ computed for every spatial frequency (from various DCT subbands) and orientation band (for cortex filters) at all spatial location $\mathbf{n}$ into a single distortion measure [SJ89, Wat93]. Daly [Dal92, Dal93] used a psychometric

function to compute the detection probability for each cortex band. Using error pooling, these detection probabilities are combined into a single error image, which describes the overall probability of detecting an error for every pixel in the image.

An example of error pooling using the Minkowski metric can be expressed as

$$P(\mathbf{n}) = \left(\sum_{\mathbf{k}} \left| \frac{C(\mathbf{k}, \mathbf{n}) - \hat{C}(\mathbf{k}, \mathbf{n})}{T(\mathbf{k}, \mathbf{n})} \right|^\beta \right)^{1/\beta}, \quad (2.38)$$

where $\hat{C}(\mathbf{k}, \mathbf{n})$ is the coded $\mathbf{k}$ th DCT-II coefficient located at $\mathbf{n}$, and β is a constant. For summation across frequency band, it is found that $\beta \approx 4$ [Wat82, GRN78, Leg78a, Leg78b, RG81, PAW93b, RAW97]. By summing all the errors in (2.38) over $\mathbf{n}$, a single value describing the distortion of an image is then obtained. For spatial error pooling over $\mathbf{n}$, several values of β have been chosen. Teo and Heeger [TH94b], Lubin [Lub93, Lub95], and Watson [Wat93] adopted β as 2, 2.4, and 4, respectively. Alternatively, this pooling process can be performed over $\mathbf{n}$, followed by over frequency bands [Wat93].

Spatial error pooling is usually carried out across the entire image to compute a single distortion measure. However, any subset of dimensions of the image can be used too. One example of this localized pooling is found in [HK02]. Höntsch and Karam [HK02] performed spatial summing within the foveal region $F(\mathbf{k}, \mathbf{n})$.

The foveal distortion is similar to (2.38), and is given as

$$P_F(\mathbf{k}, \mathbf{n}) = \left(\sum_{(\mathbf{k}', \mathbf{n}') \in F(\mathbf{k}, \mathbf{n})} \left| \frac{C(\mathbf{k}', \mathbf{n}') - \hat{C}(\mathbf{k}', \mathbf{n}')}{T_b(\mathbf{k}', \mathbf{n}')} \right|^{\beta_F} \right)^{1/\beta_F}, \quad (2.39)$$

where $\beta_F = 4$. Using the foveal distortion $P_F(\mathbf{k}, \mathbf{n})$, the distortion for $\mathbf{k}$ th DCT-II coefficient is computed as

$$P_F(\mathbf{k}) = \max_{\mathbf{n}} \{P_F(\mathbf{k}, \mathbf{n})\}, \quad (2.40)$$

and the single distortion measure of the image becomes

$$P_F = \max_{\mathbf{k}} \{P_F(\mathbf{k})\}. \quad (2.41)$$

2.6 Conversion between JND Models from Subband and Pixel

Chou and Li [CL95] used a modulation transfer function (MTF) derived in [MS74] to convert their pixel-based JND model to an equivalent JND model in the subband domain. Using the MTF, they derived a set of weights to decompose a JND profile (in pixel domain) of an image into subband JND profiles. Assuming the JND profile is divided into B equal subbands, let $t(\mathbf{x})$ and $t_b(b_i, \mathbf{s})$ denote the JND profile of an image at $\mathbf{x}$, and its b_i th subband JND profiles at $\mathbf{s}$, respectively, where $\mathbf{s} = (s_1, s_2)$, $s_1 = 0, 1, \dots, (H/\sqrt{B})-1$, $s_2 = 0, 1, \dots, (W/\sqrt{B})-1$, and $b_i = 0, 1, \dots, B-1$. The subband JND profiles are estimated by

$$t_b(b_i, \mathbf{s}) = \sqrt{w_B(b_i) \sum_{p_1=0}^{\sqrt{B}-1} \sum_{p_2=0}^{\sqrt{B}-1} t^2(s_1\sqrt{B} + p_1, s_2\sqrt{B} + p_2)}, \quad (2.42)$$

where $w_B(b_i)$ is the weighting factor derived from a MTF.

On the other hand, Zhang *et al.* [ZLX08] derived a pixel-based JND model based on the DCT subband JND model. By summing the JND threshold of the DCT subbands within an $N \times N$ DCT-II block, the pixel-based JND threshold $t'(\mathbf{x})$ is obtained:

$$t'(n_1N + p_1, n_2N + p_2) = \sum_{k_1=0}^{N-1} \sum_{k_2=0}^{N-1} \alpha_{k_1} \alpha_{k_2} T'(\mathbf{k}, \mathbf{n}) \cos\left(\frac{(2p_1+1)k_1\pi}{2N}\right) \cos\left(\frac{(2p_2+1)k_2\pi}{2N}\right),$$

for $p_1, p_2 = 0, 1, \dots, N-1$,

(2.43)

where

$$T'(\mathbf{k}, \mathbf{n}) = \begin{cases} \text{sign}(C(\mathbf{k}, \mathbf{n}))T(\mathbf{k}, \mathbf{n}), & \text{if } |C(\mathbf{k}, \mathbf{n})| \geq T(\mathbf{k}, \mathbf{n}), \\ 0, & \text{otherwise,} \end{cases} \quad (2.44)$$

and $\text{sign}(C(\mathbf{k}, \mathbf{n}))$ returns the sign of $C(\mathbf{k}, \mathbf{n})$. The sign of $T(\mathbf{k}, \mathbf{n})$ is used to avoid possible discontinuity in $t'(\mathbf{x})$ at the boundaries of each $N \times N$ DCT-II block.

2.7 Conclusions

In this chapter, we reviewed three properties of the HVS, namely, CSF, luminance adaptation, and intra- and inter-band contrast masking. These properties play important roles in the design of the image quality metric and computational model for JND based on HVS. Due to the high popularity of DCT in many image and video standards, this chapter mainly focuses on computational models for JND that are based on DCT subbands.

It is known that DCT does not match the channel decomposition mechanism of the HVS. To mitigate the issues arise from the mismatch of frequency decomposition of the HVS and DCT, Tran and Safranek [TS96] introduced a mapping from DCT-II coefficients to the cortex bands. Section 2.1 introduced the cortex filters, and reviewed the mapping of DCT-II coefficients to the cortex bands. Section 2.2 presented a widely adopted CSF proposed by Ahumada and Peterson [AP92]. The detection threshold derived from the CSF describes the sensitivity of the HVS for each spatial frequency. Since no masking is considered in the measurement of the CSF, the visibility threshold from the CSF is commonly known as the base detection threshold.

Variation to the base detection threshold is attributed by the luminance adaptation and contrast masking, which were reviewed in Section 2.3 and 2.4, respectively. Luminance adaptation refers to the variation of the base detection threshold with respect to the local luminance. Experimental results from [SJ89, CL95] indicated that the contrast sensitivity of the HVS cannot be fully explained by the Weber's law, and they found that the HVS is most sensitive in the mid luminance levels. Many image quality metrics and PICs estimated the local luminance from the DC DCT-II coefficient, or directly from the pixels of an image.

Two forms of contrast masking, namely, the intra- and inter-band contrast masking were described in Section 2.4. Most PICs have accounted for intra-band contrast masking due to its simple formulation; however more accurate representation of the JND threshold should include the inter-band contrast masking. Two estimation of the inter-band contrast masking using block classification and cortex filtering were shown in Section 2.4. In Section 2.5, we discussed how a PIC uses a single distortion measure or distortion map from the error pooling stage to determine the permissible compression for an entire image (using a single distortion measure) or different regions of the image (using a distortion map) at a predefined image quality.

Since the JND threshold in different domains (pixel or subband) is determined by the same mechanisms of the HVS, it should be possible to convert the JND threshold from the pixel domain to subband domain and vice versa. Section 2.6 presents an overview of the techniques to perform such conversion. Such an interpretation provides a convenient way for formulating the JND threshold from

the pixel domain, and effectively applying them to another domain such as DCT subband, as we shall see in Chapter 4.

This chapter reviewed the mechanisms of the HVS that are applied in many image quality metrics and PICs, we shall discuss how these mechanisms of the HVS can be applied for image compression and image transcoding in Chapters 4 and 5, respectively.

Chapter 3

Subband Discrete Cosine Transform

Karhunen-Loève Transform (KLT) was established to be an optimal transform from a statistical viewpoint [AS69, RD70]. However, the signal dependence of KLT, and the lack of a fast transform algorithm have limited the usage of KLT as a practical tool in signal processing. On the other hand, it has been shown that the DCT-II approaches the KLT in decorrelating a stationary Markov-I signal [DR58, Jai79]. It is clear that the DCT-II is widely adopted in many image and video standards, which might be reasoned that many real signals behave statistically similar to a stationary Markov-I signal. Further contribution to the popularity of the DCT-II is its vast availability of fast computational structure [LLM89, WP89a, WP89b, DG90, CH91, CL91, CYL91, WP91, CL92, FW92, CL93, PD96, HWH98, HW99, BR00].

It is well known that block-based image and video compression produces blocking artifacts at low bit-rates. An efficient transform referred as the SBDCT is proposed by Jung *et al.* [JMM96], which reduces blocking artifacts and achieves faster computation. Even though SBDCT and DCT-II are mathematically equivalent, they possess quite different computational structures. SBDCT decomposes an $N \times N$ input sequence into subbands before any transformation takes place, whereas DCT-II operates directly on the $N \times N$ input sequence. Jung *et al.* approximated the DCT-II by only operating on the dominant subband of the input sequence in SBDCT. This approximation of DCT-II requires significantly lower computational cost without introducing significant degradation in the

compressed image, and produces lesser block artifacts when bit-rate of the compressed images falls lower than 0.5 bpp. In [HH04], Hossen and Heute analyzed the aliased components of such approximation of DCT-II.

Since SBDCT is mathematically equivalent to the DCT-II, it is possible to design an image encoder using SBDCT, and retaining compatibility with the DCT-based decoder (e.g. JPEG decoder [ISO94]). On this note, this chapter proposes several variations of image encoders based on the SBDCT. This chapter starts with the derivation of the SBDCT in Section 3.1. The FSA and seven approximations of the DCT-II based on SBDCT are presented together with the analysis of their computational cost in Section 3.2. Section 3.3 discusses several schemes of adaptive SBDCT. These adaptive schemes take into account of the aliased components found in the SBDCT to obtain a good approximation of DCT-II with respect to the energy distribution of an $N \times N$ input sequence. A discussion of the simulation results is presented in Section 3.4. Finally, our conclusions for this chapter are given in Section 3.5.

3.1 Derivation of Subband Discrete Cosine Transform

The one-dimensional (1-D) DCT-II of the n_1 th N -point sequence, $c(p_1, n_1)$, is given by

$$C(k_1, n_1) = \alpha_{k_1} \sum_{p_1=0}^{N-1} c(p_1, n_1) \cos\left(\frac{(2p_1+1)k_1\pi}{2N}\right). \quad (3.1)$$

By defining two subsequences $c(0, q_1, n_1) = 0.5(c(2q_1, n_1) + c(2q_1+1, n_1))$ and $c(1, q_1, n_1) = 0.5(c(2q_1, n_1) - c(2q_1+1, n_1))$, where $p_1 = 0, 1, \dots, N-1$, and $q_1 = 0, 1, \dots, N/2-1$, we can decompose (3.1) into

$$\begin{aligned}
C(k_1, n_1) &= 2\alpha_{k_1} W_c^{k_1/2N} \sum_{q_1=0}^{N'-1} c(0, q_1, n_1) W_c^{(2q_1+1)k_1/N} \\
&\quad + 2\alpha_{k_1} W_s^{k_1/2N} \sum_{q_1=0}^{N'-1} c(1, q_1, n_1) W_s^{(2q_1+1)k_1/N},
\end{aligned} \tag{3.2}$$

where $W_c^{a/N} = \cos(a\pi/N)$, $W_s^{a/N} = \sin(a\pi/N)$, and $N' = N/2$. From (3.2), we see that the N -point DCT-II can be further decomposed into an N' -point DCT-II, and another transform having sine basis vector. This transform is referred as the subband discrete sine transform (DST). Let $C(0, k_1, n_1)$ and $C(1, k_1, n_1)$ denote the 1-D subband transforms of $c(0, q_1, n_1)$ and $c(1, q_1, n_1)$, respectively. These subband transforms are defined as

$$\begin{aligned}
C(0, k_1, n_1) &= \sum_{q_1=0}^{N'-1} c(0, q_1, n_1) W_c^{(2q_1+1)k_1/N}, \\
C(1, k_1, n_1) &= \sum_{q_1=0}^{N'-1} c(1, q_1, n_1) W_s^{(2q_1+1)k_1/N}.
\end{aligned} \tag{3.3}$$

Since the 2-D DCT-II kernel is separable, the computation of the DCT-II can be realized by row-column decomposition, which sequentially performs 1-D DCT-II on the rows and columns of the input sequence. Using row-column decomposition to decompose the 1-D DCT-II, we can re-express the 2-D DCT-II of the $\mathbf{n}$ th $N \times N$ sub-image of an image, $c(\mathbf{p}, \mathbf{n})$, as

$$\begin{aligned}
C(\mathbf{k}, \mathbf{n}) &= \alpha_{k_1} \alpha_{k_2} \sum_{p_1=0}^{N-1} \sum_{p_2=0}^{N-1} c(\mathbf{p}, \mathbf{n}) W_c^{(2p_1+1)k_1/2N} W_c^{(2p_2+1)k_2/2N} \\
&= 4\alpha_{k_1} \alpha_{k_2} W_c^{k_1/2N} W_c^{k_2/2N} \sum_{q_1=0}^{N'-1} \sum_{q_2=0}^{N'-1} c(0, \mathbf{q}, \mathbf{n}) W_c^{(2q_1+1)k_1/N} W_c^{(2q_2+1)k_2/N} \\
&\quad + 4\alpha_{k_1} \alpha_{k_2} W_c^{k_1/2N} W_s^{k_2/2N} \sum_{q_1=0}^{N'-1} \sum_{q_2=0}^{N'-1} c(1, \mathbf{q}, \mathbf{n}) W_c^{(2q_1+1)k_1/N} W_s^{(2q_2+1)k_2/N} \\
&\quad + 4\alpha_{k_1} \alpha_{k_2} W_s^{k_1/2N} W_c^{k_2/2N} \sum_{q_1=0}^{N'-1} \sum_{q_2=0}^{N'-1} c(2, \mathbf{q}, \mathbf{n}) W_s^{(2q_1+1)k_1/N} W_c^{(2q_2+1)k_2/N} \\
&\quad + 4\alpha_{k_1} \alpha_{k_2} W_s^{k_1/2N} W_s^{k_2/2N} \sum_{q_1=0}^{N'-1} \sum_{q_2=0}^{N'-1} c(3, \mathbf{q}, \mathbf{n}) W_s^{(2q_1+1)k_1/N} W_s^{(2q_2+1)k_2/N},
\end{aligned} \tag{3.4}$$

where $\mathbf{p} = (p_1, p_2)$, $\mathbf{q} = (q_1, q_2)$, $c(b_i, \mathbf{q}, \mathbf{n})$ represents the subbands of $c(\mathbf{p}, \mathbf{n})$, and $b_i = 0, 1, \dots, 3$, $p_1, p_2 = 0, 1, \dots, N-1$, and $q_1, q_2 = 0, 1, \dots, N'-1$. Equation (3.4) is referred as the fullband SBDCT, and the subbands of $c(\mathbf{p}, \mathbf{n})$ are computed as

$$\begin{aligned} c(0, \mathbf{q}, \mathbf{n}) &= \frac{1}{4} \sum_{m_1=0}^1 \sum_{m_2=0}^1 c(2q_1 + m_1, 2q_2 + m_2, \mathbf{n}), \\ c(1, \mathbf{q}, \mathbf{n}) &= \frac{1}{4} \sum_{m_1=0}^1 \sum_{m_2=0}^1 (-1)^{m_2} c(2q_1 + m_1, 2q_2 + m_2, \mathbf{n}), \\ c(2, \mathbf{q}, \mathbf{n}) &= \frac{1}{4} \sum_{m_1=0}^1 \sum_{m_2=0}^1 (-1)^{m_1} c(2q_1 + m_1, 2q_2 + m_2, \mathbf{n}), \\ c(3, \mathbf{q}, \mathbf{n}) &= \frac{1}{4} \sum_{m_1=0}^1 \sum_{m_2=0}^1 (-1)^{m_1+m_2} c(2q_1 + m_1, 2q_2 + m_2, \mathbf{n}). \end{aligned} \quad (3.5)$$

Let $C(b_i, \mathbf{k}, \mathbf{n})$ denote the 2-D subband transform of $c(b_i, \mathbf{q}, \mathbf{n})$. These 2-D subband transforms are given by

$$\begin{aligned} C(0, \mathbf{k}, \mathbf{n}) &= \sum_{q_1=0}^{N'-1} \sum_{q_2=0}^{N'-1} c(0, \mathbf{q}, \mathbf{n}) W_c^{(2q_1+1)k_1/N} W_c^{(2q_2+1)k_2/N}, \\ C(1, \mathbf{k}, \mathbf{n}) &= \sum_{q_1=0}^{N'-1} \sum_{q_2=0}^{N'-1} c(1, \mathbf{q}, \mathbf{n}) W_c^{(2q_1+1)k_1/N} W_s^{(2q_2+1)k_2/N}, \\ C(2, \mathbf{k}, \mathbf{n}) &= \sum_{q_1=0}^{N'-1} \sum_{q_2=0}^{N'-1} c(2, \mathbf{q}, \mathbf{n}) W_s^{(2q_1+1)k_1/N} W_c^{(2q_2+1)k_2/N}, \\ C(3, \mathbf{k}, \mathbf{n}) &= \sum_{q_1=0}^{N'-1} \sum_{q_2=0}^{N'-1} c(3, \mathbf{q}, \mathbf{n}) W_s^{(2q_1+1)k_1/N} W_s^{(2q_2+1)k_2/N}. \end{aligned} \quad (3.6)$$

Equation (3.4) can be compactly expressed as

$$\begin{aligned} C(\mathbf{k}, \mathbf{n}) &= 4\alpha_{k_1} \alpha_{k_2} W_c^{k_1/2N} W_c^{k_2/2N} C(0, \mathbf{k}, \mathbf{n}) + 4\alpha_{k_1} \alpha_{k_2} W_c^{k_1/2N} W_s^{k_2/2N} C(1, \mathbf{k}, \mathbf{n}) \\ &\quad + 4\alpha_{k_1} \alpha_{k_2} W_s^{k_1/2N} W_c^{k_2/2N} C(2, \mathbf{k}, \mathbf{n}) + 4\alpha_{k_1} \alpha_{k_2} W_s^{k_1/2N} W_s^{k_2/2N} C(3, \mathbf{k}, \mathbf{n}) \quad (3.7) \\ &= \sum_{b_i=0}^3 W_{wn}(b_i, \mathbf{k}) C(b_i, \mathbf{k}, \mathbf{n}), \end{aligned}$$

where $W_{wn}(0, \mathbf{k})$, $W_{wn}(1, \mathbf{k})$, $W_{wn}(2, \mathbf{k})$, and $W_{wn}(3, \mathbf{k})$ are $4\alpha_{k_1} \alpha_{k_2} W_c^{k_1/2N} W_c^{k_2/2N}$,

$4\alpha_{k_1} \alpha_{k_2} W_c^{k_1/2N} W_s^{k_2/2N}$, $4\alpha_{k_1} \alpha_{k_2} W_s^{k_1/2N} W_c^{k_2/2N}$, and $4\alpha_{k_1} \alpha_{k_2} W_s^{k_1/2N} W_s^{k_2/2N}$, respectively.

If any subband $c(b_i, \mathbf{q}, \mathbf{n})$ has negligible energy, we can avoid computing its subband transform to reduce computational cost. Assuming $c(0, \mathbf{q}, \mathbf{n})$ is the dominant subband of $c(\mathbf{p}, \mathbf{n})$, a reasonable approximation of (3.7) is obtained by only evaluating $W_{wn}(0, \mathbf{k})C(0, \mathbf{k}, \mathbf{n})$. This approximation produces a reconstructed image of reasonable quality while significantly reduces computational cost [JMM96].

An illustration of the fullband SBDCT is presented in Fig. 3.1. Clearly, the structure of the SBDCT is well suited for parallel processing [TGW05b] since the four subbands can be independently processed. It is noted that $c(0, \mathbf{q}, \mathbf{n})$ can be further decomposed by substituting $c(\mathbf{p}, \mathbf{n})$ in (3.5) with $c(0, \mathbf{q}, \mathbf{n})$. However, a single level of subband decomposition is chosen due to its simplicity in implementation.

Fig. 3.1 Four subband transforms of $c(b_i, \mathbf{q}, \mathbf{n})$ in SBDCT.

3.2 Fast Subband Discrete Cosine Transform

The 1-D subband transforms in (3.3) can be rewritten as

$$\mathbf{C}_{0,n_1}^{\text{1D}} = \mathbf{T}_c \begin{bmatrix} c(0,0,n_1) & c(0,1,n_1) & c(0,2,n_1) & c(0,3,n_1) \end{bmatrix}^t \quad \text{and} \quad \mathbf{C}_{1,n_1}^{\text{1D}} = \mathbf{T}_s \begin{bmatrix} c(1,0,n_1) \\ c(1,1,n_1) \\ c(1,2,n_1) \\ c(1,3,n_1) \end{bmatrix}^t,$$

where t is the transpose operator; $C(0, k_1, n_1)$

and $C(1, k_1, n_1)$ denote the elements of $\mathbf{C}_{0, n_1}^{1D}$ and $\mathbf{C}_{1, n_2}^{1D}$, respectively; $\mathbf{T}_c$ and $\mathbf{T}_s$ are the basis vectors of an N' -point subband DCT and subband DST, respectively. Assuming $N = 8$ and $N' = 4$, the basis vectors $\mathbf{T}_c$ and $\mathbf{T}_s$ are given in the following matrices:

$$\mathbf{T}_c = \begin{bmatrix} 1 & W_c^{1/8} & W_c^{1/4} & W_c^{3/8} & 0 & -W_c^{3/8} & -W_c^{1/4} & -W_c^{1/8} \\ 1 & W_c^{3/8} & -W_c^{1/4} & -W_c^{1/8} & 0 & W_c^{1/8} & W_c^{1/4} & -W_c^{3/8} \\ 1 & -W_c^{3/8} & -W_c^{1/4} & W_c^{1/8} & 0 & -W_c^{1/8} & W_c^{1/4} & W_c^{3/8} \\ 1 & -W_c^{1/8} & W_c^{1/4} & -W_c^{3/8} & 0 & W_c^{3/8} & -W_c^{1/4} & W_c^{1/8} \end{bmatrix}^t, \quad (3.8)$$

$$\mathbf{T}_s = \begin{bmatrix} 0 & W_c^{3/8} & W_c^{1/4} & W_c^{1/8} & 1 & W_c^{1/8} & W_c^{1/4} & W_c^{3/8} \\ 0 & W_c^{1/8} & W_c^{1/4} & -W_c^{3/8} & -1 & -W_c^{3/8} & W_c^{1/4} & W_c^{1/8} \\ 0 & W_c^{1/8} & -W_c^{1/4} & -W_c^{3/8} & 1 & -W_c^{3/8} & -W_c^{1/4} & W_c^{1/8} \\ 0 & W_c^{3/8} & -W_c^{1/4} & W_c^{1/8} & -1 & W_c^{1/8} & -W_c^{1/4} & W_c^{3/8} \end{bmatrix}^t,$$

where the first four rows of $\mathbf{T}_c$ is the basis vector of a 4-point DCT-II. Odd and even symmetry of the basis vectors about the fifth row are observed in $\mathbf{T}_c$ and $\mathbf{T}_s$, respectively. Let $C(b_i, \mathbf{k}, \mathbf{n})$ denote the elements of $\mathbf{C}_{b_i, \mathbf{n}}^{2D}$, where $b_i = 0, 1, 2, 3$. These 2-D subband transforms can be written as matrix multiplication of $\mathbf{C}_{0, \mathbf{n}}^{2D} = \mathbf{T}_c \mathbf{c}_{0, \mathbf{n}} \mathbf{T}_c^t$, $\mathbf{C}_{1, \mathbf{n}}^{2D} = \mathbf{T}_c \mathbf{c}_{1, \mathbf{n}} \mathbf{T}_s^t$, $\mathbf{C}_{2, \mathbf{n}}^{2D} = \mathbf{T}_s \mathbf{c}_{2, \mathbf{n}} \mathbf{T}_c^t$, and $\mathbf{C}_{3, \mathbf{n}}^{2D} = \mathbf{T}_s \mathbf{c}_{3, \mathbf{n}} \mathbf{T}_s^t$. The elements of $\mathbf{c}_{b_i, \mathbf{n}}$ are given by $c(b_i, \mathbf{q}, \mathbf{n})$, where $q_1, q_2 = 0, 1, 2, 3$. Let $\mathbf{c}_\mathbf{n}$ denote an 8×8 matrix of the $\mathbf{n}$ th sub-image of an image, and its DCT coefficients is denoted by $\mathbf{C}_\mathbf{n}$, where the elements of $\mathbf{c}_\mathbf{n}$ and $\mathbf{C}_\mathbf{n}$ are given by $c(\mathbf{p}, \mathbf{n})$, and $C(\mathbf{k}, \mathbf{n})$, respectively. From (3.4) and (3.5), we can express the fullband SBDCT as

$$\begin{aligned} \mathbf{C}_\mathbf{n} &= \mathbf{D} \mathbf{D}_c \mathbf{C}_{0, \mathbf{n}}^{2D} \mathbf{D}_c^t \mathbf{D} + \mathbf{D} \mathbf{D}_c \mathbf{C}_{1, \mathbf{n}}^{2D} \mathbf{D}_s^t \mathbf{D}^t + \mathbf{D} \mathbf{D}_s \mathbf{C}_{2, \mathbf{n}}^{2D} \mathbf{D}_c^t \mathbf{D}^t + \mathbf{D} \mathbf{D}_s \mathbf{C}_{3, \mathbf{n}}^{2D} \mathbf{D}_s^t \mathbf{D}^t \\ &= \mathbf{D} \begin{bmatrix} \mathbf{D}_c \mathbf{T}_c & \mathbf{D}_s \mathbf{T}_s \end{bmatrix} \begin{bmatrix} \mathbf{c}_{0, \mathbf{n}} & \mathbf{c}_{1, \mathbf{n}} \\ \mathbf{c}_{2, \mathbf{n}} & \mathbf{c}_{3, \mathbf{n}} \end{bmatrix} \begin{bmatrix} \mathbf{T}_c^t \mathbf{D}_c^t \\ \mathbf{T}_s^t \mathbf{D}_s^t \end{bmatrix} \mathbf{D}^t \\ &= \mathbf{D} \begin{bmatrix} \mathbf{D}_c & \mathbf{D}_s \end{bmatrix} \begin{bmatrix} \mathbf{T}_c & \mathbf{0}_{8 \times 4} \\ \mathbf{0}_{8 \times 4} & \mathbf{T}_s \end{bmatrix} \mathbf{P}_2 (\mathbf{I}_4 \otimes \mathbf{P}_1) \mathbf{c}_\mathbf{n} (\mathbf{I}_4 \otimes \mathbf{P}_1)^t \mathbf{P}_2^t \begin{bmatrix} \mathbf{T}_c^t & \mathbf{0}_{4 \times 8} \\ \mathbf{0}_{4 \times 8} & \mathbf{T}_s^t \end{bmatrix} \begin{bmatrix} \mathbf{D}_c^t \\ \mathbf{D}_s^t \end{bmatrix} \mathbf{D}^t \end{aligned} \quad (3.9)$$

where $\otimes$ denotes the Kronecker product; $\mathbf{D}$ and $\mathbf{I}_n$ denote an $N \times N$ diagonal matrix and an $n \times n$ identity matrix, respectively, and the permutation matrices $\mathbf{P}_1$, and $\mathbf{P}_2$ are given as

$$\mathbf{P}_1 = \begin{bmatrix} 0.5 & 0.5 \\ 0.5 & -0.5 \end{bmatrix}, \quad \mathbf{P}_2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (3.10)$$

The elements of the $N \times N$ diagonal matrices $\mathbf{D}$, $\mathbf{D}_c$, and $\mathbf{D}_s$ are $\text{diag}\{\sqrt{1/2}, 1, 1, \dots, 1\}$, $\text{diag}\{1, W_c^{1/16}, W_c^{2/16}, \dots, W_c^{7/16}\}$, and $\text{diag}\{0, W_s^{1/16}, W_s^{2/16}, \dots, W_s^{7/16}\}$, respectively, where $\text{diag}\{\cdot\}$ denotes a diagonal matrix.

Each coefficient of $\mathbf{c}_{b_i, \mathbf{n}}$ requires 3 additions to compute, therefore each subband $\mathbf{c}_{b_i, \mathbf{n}}$ requires $3 \times 16 = 48$ additions if we compute them individually. The computational cost to compute four subbands becomes 192 additions. With $A(\mathbf{n}) = c(b_i, 2q_1, 2q_2, \mathbf{n}) \pm c(b_i, 2q_1, 2q_2 + 1, \mathbf{n})$, and $B(\mathbf{n}) = c(b_i, 2q_1 + 1, 2q_2, \mathbf{n}) \pm c(b_i, 2q_1 + 1, 2q_2 + 1, \mathbf{n})$, a total of 128 additions instead of 192 additions is required to compute the four subbands.

From (3.8), it is clear that we can obtain $\mathbf{T}_s$ from $\mathbf{T}_c$ through permutation of its rows and columns. Taking this into consideration, we can express $\mathbf{T}_s$ as $\mathbf{P}_3 \mathbf{T}_c \mathbf{P}_4$

where $\mathbf{P}_3 = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \otimes \mathbf{I}_4$ and $\mathbf{P}_4 = \mathbf{I}_2 \otimes \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$. By obtaining $\mathbf{T}_s$ from $\mathbf{T}_c$, we can

compute the fullband SBDCT by using four 2-D DCT-II. Thus, we can rewrite (3.9) as

$$\begin{aligned} \mathbf{C}_n = & \mathbf{D} \begin{bmatrix} \mathbf{D}_c & \mathbf{D}_s \end{bmatrix} \begin{bmatrix} \mathbf{T}_c & \mathbf{0}_{8 \times 4} \\ \mathbf{0}_{8 \times 4} & \mathbf{P}_3 \mathbf{T}_c \mathbf{P}_4 \end{bmatrix} \mathbf{P}_2 (\mathbf{I}_4 \otimes \mathbf{P}_1) \mathbf{c}_n (\mathbf{I}_4 \otimes \mathbf{P}_1)^t \mathbf{P}_2^t \\ & \cdot \begin{bmatrix} \mathbf{T}_c^t & \mathbf{0}_{4 \times 8} \\ \mathbf{0}_{4 \times 8} & \mathbf{P}_4^t \mathbf{T}_c^t \mathbf{P}_3^t \end{bmatrix} \begin{bmatrix} \mathbf{D}_c^t \\ \mathbf{D}_s^t \end{bmatrix} \mathbf{D}^t. \end{aligned} \quad (3.11)$$

Equation (3.11) requires significant multiplications due to the diagonal matrices.

However, several diagonal matrices can be combined, and leads to the following:

$$\begin{aligned} \mathbf{C}_n = & \mathbf{D} \mathbf{D}_c \mathbf{T}_c \mathbf{c}_{0,n} \mathbf{T}_c^t \mathbf{D}_c^t \mathbf{D}^t + \mathbf{D} \mathbf{D}_c \mathbf{T}_c \mathbf{c}_{1,n} \mathbf{P}_4^t \mathbf{T}_c^t \mathbf{P}_3^t \mathbf{D}_s^t \mathbf{D}^t + \mathbf{D} \mathbf{D}_s \mathbf{P}_3 \mathbf{T}_c \mathbf{P}_4 \mathbf{c}_{2,n} \mathbf{T}_c^t \mathbf{D}_c^t \mathbf{D}^t \\ & + \mathbf{D} \mathbf{D}_s \mathbf{P}_3 \mathbf{T}_c \mathbf{P}_4 \mathbf{c}_{3,n} \mathbf{P}_4^t \mathbf{T}_c^t \mathbf{P}_3^t \mathbf{D}_s^t \mathbf{D}^t \\ = & \mathbf{D} \mathbf{D}_c \left(\mathbf{T}_c \mathbf{c}_{0,n} \mathbf{T}_c^t \right) \mathbf{D}_c^t \mathbf{D}^t + \mathbf{D} \mathbf{D}_c \left(\mathbf{T}_c \mathbf{c}_{1,n}' \mathbf{T}_c^t \mathbf{P}_3^t \right) \mathbf{D}_s^t \mathbf{D}^t \\ & + \mathbf{D} \mathbf{D}_s \left(\mathbf{P}_3 \mathbf{T}_c \mathbf{c}_{2,n}' \mathbf{T}_c^t \right) \mathbf{D}_c^t \mathbf{D}^t + \mathbf{D} \mathbf{D}_s \left(\mathbf{P}_3 \mathbf{T}_c \mathbf{c}_{3,n}' \mathbf{T}_c^t \mathbf{P}_3^t \right) \mathbf{D}_s^t \mathbf{D}^t \\ = & \mathbf{W}_{cc} \cdot * \left(\mathbf{T}_c \mathbf{c}_{0,n} \mathbf{T}_c^t \right) + \mathbf{W}_{cs} \cdot * \left(\mathbf{T}_c \mathbf{c}_{1,n}' \mathbf{T}_c^t \mathbf{P}_3^t \right) + \mathbf{W}_{sc} \cdot * \left(\mathbf{P}_3 \mathbf{T}_c \mathbf{c}_{2,n}' \mathbf{T}_c^t \right) \\ & + \mathbf{W}_{ss} \cdot * \left(\mathbf{P}_3 \mathbf{T}_c \mathbf{c}_{3,n}' \mathbf{T}_c^t \mathbf{P}_3^t \right), \end{aligned} \quad (3.12)$$

where $\cdot *$ denotes element-by-element multiplication of matrices; $\mathbf{c}_{1,n}'$, $\mathbf{c}_{2,n}'$, and $\mathbf{c}_{3,n}'$ are $\mathbf{c}_{1,n} \mathbf{P}_4^t$, $\mathbf{P}_4 \mathbf{c}_{2,n}$, and $\mathbf{P}_4 \mathbf{c}_{3,n} \mathbf{P}_4^t$, respectively. Using (3.12), the four subband transforms in fullband SBDCT are obtained using $\mathbf{T}_c$, instead of combinations of $\mathbf{T}_c$ and $\mathbf{T}_s$. Therefore, by designing one fast algorithm based on $\mathbf{T}_c$, we are able to efficiently compute the four subband transforms of $\mathbf{C}_n$.

Jung *et al.* [JMM96] observed the odd and even symmetry about the fifth row in $\mathbf{T}_c$ and $\mathbf{T}_s$, respectively. Hence, there is no need to compute all the transform coefficients of subband DCT and DST. Let $\mathbf{T}_c'$ denote the first four rows of $\mathbf{T}_c$ and $\mathbf{T}_c'$ is the basis vector of the 4-point DCT-II. Hence, the 16 unique transform coefficients of $\mathbf{T}_c \mathbf{c}_{0,n} \mathbf{T}_c^t$, which is obtained with $\mathbf{T}_c' \mathbf{c}_{0,n} (\mathbf{T}_c')^t$, is the 4×4 DCT-II of $\mathbf{c}_{0,n}$. Likewise for $\mathbf{T}_c \mathbf{c}_{1,n}' \mathbf{T}_c^t$, $\mathbf{T}_c \mathbf{c}_{2,n}' \mathbf{T}_c^t$, and $\mathbf{T}_c \mathbf{c}_{3,n}' \mathbf{T}_c^t$, we can compute the 16

unique transform coefficients using $\mathbf{T}'_c \mathbf{c}'_{1,n} (\mathbf{T}'_c)^t$, $\mathbf{T}'_c \mathbf{c}'_{2,n} (\mathbf{T}'_c)^t$, and $\mathbf{T}'_c \mathbf{c}'_{3,n} (\mathbf{T}'_c)^t$, respectively, and apply the symmetry of these transform coefficients to obtain the transform coefficients of these subband transforms. By extending Jung's observation of the symmetry in $\mathbf{T}_c$ and $\mathbf{T}_s$, the symmetry properties of the four 2-D subband transforms are stated as

$$C(0, \mathbf{k}, \mathbf{n}) = \begin{cases} \mathbf{T}'_c \mathbf{c}'_{0,n} (\mathbf{T}'_c)^t, & k_1 < 4, k_2 < 4, \\ 0, & k_1 = 4, \\ 0, & k_2 = 4, \\ -C(0, k_1, 8 - k_2, \mathbf{n}), & k_1 < 4, 5 \leq k_2 < 8, \\ -C(0, 8 - k_1, k_2, \mathbf{n}), & 5 \leq k_1 < 8, k_2 < 4, \\ C(0, 8 - k_1, 8 - k_2, \mathbf{n}), & 5 \leq k_1 < 8, 5 \leq k_2 < 8, \end{cases} \quad (3.13)$$

$$C(1, \mathbf{k}, \mathbf{n}) = \begin{cases} \mathbf{T}'_c \mathbf{c}'_{1,n} \mathbf{T}'_c \mathbf{P}'_3, & k_1 < 4, 1 \leq k_2 < 5, \\ 0, & k_1 = 4, \\ 0, & k_2 = 0, \\ C(1, k_1, 8 - k_2, \mathbf{n}), & k_1 < 4, 5 \leq k_2 < 8, \\ -C(1, 8 - k_1, k_2, \mathbf{n}), & 5 \leq k_1 < 8, 1 \leq k_2 < 5, \\ -C(1, 8 - k_1, 8 - k_2, \mathbf{n}), & 5 \leq k_1 < 8, 5 \leq k_2 < 8, \end{cases} \quad (3.14)$$

$$C(2, \mathbf{k}, \mathbf{n}) = \begin{cases} \mathbf{P}_3 \mathbf{T}'_c \mathbf{c}'_{2,n} \mathbf{T}'_c, & 1 \leq k_1 < 5, k_2 < 4, \\ 0, & k_1 = 0, \\ 0, & k_2 = 4, \\ -C(2, k_1, 8 - k_2, \mathbf{n}), & 1 \leq k_1 < 5, 5 \leq k_2 < 8, \\ C(2, 8 - k_1, k_2, \mathbf{n}), & 5 \leq k_1 < 8, k_2 < 4, \\ -C(2, 8 - k_1, 8 - k_2, \mathbf{n}), & 5 \leq k_1 < 8, 5 \leq k_2 < 8, \end{cases} \quad (3.15)$$

and

$$C(3, \mathbf{k}, \mathbf{n}) = \begin{cases} \mathbf{P}_3 \mathbf{T}'_c \mathbf{c}'_{3,n} \mathbf{T}'_c \mathbf{P}'_3, & 1 \leq k_1 < 5, 1 \leq k_2 < 5, \\ 0, & k_1 = 0, \\ 0, & k_2 = 0, \\ C(3, k_1, 8 - k_2, \mathbf{n}), & 1 \leq k_1 < 5, 5 \leq k_2 < 8, \\ C(3, 8 - k_1, 8 - k_2, \mathbf{n}), & 5 \leq k_1 < 8, 1 \leq k_2 < 5, \\ C(3, 8 - k_1, 8 - k_2, \mathbf{n}), & 5 \leq k_1 < 8, 5 \leq k_2 < 8. \end{cases} \quad (3.16)$$

Next, we investigate the twiddle factor matrices $\mathbf{W}_{cc}$, $\mathbf{W}_{cs}$, $\mathbf{W}_{sc}$, and $\mathbf{W}_{ss}$ in (3.12). Let $W_{cc}(\mathbf{k})$, $W_{cs}(\mathbf{k})$, $W_{sc}(\mathbf{k})$, and $W_{ss}(\mathbf{k})$ denote the elements of $\mathbf{W}_{cc}$, $\mathbf{W}_{cs}$, $\mathbf{W}_{sc}$, and $\mathbf{W}_{ss}$, respectively. The elements of these twiddle factor matrices are given as

$$\begin{aligned} W_{cc}(\mathbf{k}) &= \alpha_{k_1} \alpha_{k_2} W_c^{k_1/16} W_c^{k_2/16}, \\ W_{cs}(\mathbf{k}) &= \alpha_{k_1} \alpha_{k_2} W_c^{k_1/16} W_s^{k_2/16}, \\ W_{sc}(\mathbf{k}) &= \alpha_{k_1} \alpha_{k_2} W_s^{k_1/16} W_c^{k_2/16}, \\ W_{ss}(\mathbf{k}) &= \alpha_{k_1} \alpha_{k_2} W_s^{k_1/16} W_s^{k_2/16}. \end{aligned} \quad (3.17)$$

Considering the zero elements in $\mathbf{W}_{cc} \cdot \mathbf{C}_{0,\mathbf{n}}^{2D}$, $\mathbf{W}_{cs} \cdot \mathbf{C}_{1,\mathbf{n}}^{2D}$, $\mathbf{W}_{sc} \cdot \mathbf{C}_{2,\mathbf{n}}^{2D}$, and $\mathbf{W}_{ss} \cdot \mathbf{C}_{3,\mathbf{n}}^{2D}$, these four sets of element-by-element multiplications require 196 multiplications. To further reduce the computational cost required by $\mathbf{W}_{cc}$, $\mathbf{W}_{cs}$, $\mathbf{W}_{sc}$, and $\mathbf{W}_{ss}$, we factorize the twiddle factor matrices. Let (c_m, c_a) denote the computational cost, where c_m and c_a are the number of multiplications and additions, respectively. The computational cost of 12 possible factorizations of the twiddle factor matrices are presented in Table 3.1. Some elements in $\mathbf{W}_{cc}/\mathbf{B}$, $\mathbf{W}_{cs}/\mathbf{B}$, $\mathbf{W}_{sc}/\mathbf{B}$, and $\mathbf{W}_{ss}/\mathbf{B}$, where $\mathbf{B}$ can be $\mathbf{W}_{cc}$, $\mathbf{W}_{cs}$, $\mathbf{W}_{sc}$, or $\mathbf{W}_{ss}$, may become infinity due to the zero elements in $\mathbf{B}$. This problem is avoided by replacing the zero elements in $\mathbf{B}$ with '1' to avoid dividing by zero. This replacement does not affect the result of (3.12) since a zero element in $\mathbf{W}_{cc}$, $\mathbf{W}_{cs}$, $\mathbf{W}_{sc}$, and $\mathbf{W}_{ss}$ is always accompanied by a zero element at the same location of $\mathbf{C}_{0,\mathbf{n}}^{2D}$, $\mathbf{C}_{1,\mathbf{n}}^{2D}$, $\mathbf{C}_{2,\mathbf{n}}^{2D}$, and $\mathbf{C}_{3,\mathbf{n}}^{2D}$, respectively. The updated matrices $\mathbf{W}_{cs}$, $\mathbf{W}_{sc}$, or $\mathbf{W}_{ss}$ (having zero elements) are denoted as $\mathbf{W}'_{cs}$, $\mathbf{W}'_{sc}$, or $\mathbf{W}'_{ss}$, respectively. The factorizations $\mathbf{W}_{cc}/\mathbf{B}$, $\mathbf{W}_{cs}/\mathbf{B}$, $\mathbf{W}_{sc}/\mathbf{B}$, and $\mathbf{W}_{ss}/\mathbf{B}$ may lead to a new twiddle

Table 3.1 Required number of multiplications for element-by-element multiplication of twiddle factor matrices and 2-D subband transform, where R and C denote the same element is found in the row and column of $\mathbf{W}_{cc} \cdot / \mathbf{B}$, $\mathbf{W}'_{cs} \cdot / \mathbf{B}$, $\mathbf{W}'_{sc} \cdot / \mathbf{B}$, and $\mathbf{W}'_{ss} \cdot / \mathbf{B}$, respectively.

Factorization	Computational Cost (c_m, c_a)			
	$\mathbf{B} = \mathbf{W}_{cc}$	$\mathbf{B} = \mathbf{W}'_{cs}$	$\mathbf{B} = \mathbf{W}'_{sc}$	$\mathbf{B} = \mathbf{W}'_{ss}$
$(\mathbf{W}_{cc} \cdot / \mathbf{B}) \cdot * \mathbf{C}_{0,n}^{2D}$		(26, 4) (C)	(26, 4) (R)	(42, 0)
$(\mathbf{W}_{cs} \cdot / \mathbf{B}) \cdot * \mathbf{C}_{1,n}^{2D}$	(20, 4) (C)		(42, 0)	(26, 4) (R)
$(\mathbf{W}_{sc} \cdot / \mathbf{B}) \cdot * \mathbf{C}_{2,n}^{2D}$	(20, 4) (R)	(42, 0)		(26, 4) (C)
$(\mathbf{W}_{ss} \cdot / \mathbf{B}) \cdot * \mathbf{C}_{3,n}^{2D}$	(42, 0)	(20, 4) (R)	(20, 4) (C)	

factor matrix that has the same elements in its columns or rows and are denoted as C and R in Table 3.1, respectively, and $\cdot /$ denotes element-by-element division of matrices. It is clear that the factorizations $\mathbf{W}_{ss} \cdot / \mathbf{W}_{cc}$, $\mathbf{W}_{sc} \cdot / \mathbf{W}'_{cs}$, $\mathbf{W}_{cs} \cdot / \mathbf{W}'_{sc}$, and $\mathbf{W}_{cc} \cdot / \mathbf{W}'_{ss}$ required the most multiplications in Table 3.1 and should be avoided. By selecting the factorizations that lead to the lowest number of multiplications in (3.12), we have

$$\begin{aligned}
\mathbf{C}_n &= \mathbf{W}_{cs} \cdot * (\mathbf{T}_c \mathbf{c}'_{1,n} \mathbf{T}_c' \mathbf{P}_3') + \mathbf{W}_{ss} \cdot * (\mathbf{P}_3 \mathbf{T}_c \mathbf{c}'_{3,n} \mathbf{T}_c' \mathbf{P}_3') \\
&\quad + \mathbf{W}_{cc} \cdot * (\mathbf{T}_c \mathbf{c}_{0,n} \mathbf{T}_c') + \mathbf{W}_{sc} \cdot * (\mathbf{P}_3 \mathbf{T}_c \mathbf{c}'_{2,n} \mathbf{T}_c') \\
&= \mathbf{W}_{cc} \cdot * \left\{ (\mathbf{W}_{cs} \cdot / \mathbf{W}_{cc}) \cdot * \left[(\mathbf{T}_c \mathbf{c}'_{1,n} \mathbf{T}_c' \mathbf{P}_3') + (\mathbf{W}_{ss} \cdot / \mathbf{W}'_{cs}) \cdot * (\mathbf{P}_3 \mathbf{T}_c \mathbf{c}'_{3,n} \mathbf{T}_c' \mathbf{P}_3') \right] \right. \\
&\quad \left. + \left[(\mathbf{T}_c \mathbf{c}_{0,n} \mathbf{T}_c') + (\mathbf{W}_{sc} \cdot / \mathbf{W}_{cc}) \cdot * (\mathbf{P}_3 \mathbf{T}_c \mathbf{c}'_{2,n} \mathbf{T}_c') \right] \right\}.
\end{aligned} \tag{3.18}$$

Interestingly, (3.18) does not need any modification of its twiddle matrix $\mathbf{W}_{cc}$ as there is no zero element in $\mathbf{W}_{cc}$. From (3.18), we have three new twiddle factor matrices $\mathbf{W}_{cs} \cdot / \mathbf{W}_{cc}$, $\mathbf{W}_{ss} \cdot / \mathbf{W}'_{cs}$, and $\mathbf{W}_{sc} \cdot / \mathbf{W}_{cc}$, which are denoted as $\mathbf{W}_{cs/cc}$, $\mathbf{W}_{ss/cs}$, and $\mathbf{W}_{sc/cc}$, respectively. From (3.17), we have

$$\mathbf{W}_{ss/cs} = \begin{cases} 0, & k_1 = 0, \text{ or } k_2 = 0, \\ \tan(k_1\pi/16), & \text{otherwise,} \end{cases} \quad (3.19)$$

$$\mathbf{W}_{sc/cc} = \begin{cases} \frac{\sqrt{8}}{\alpha_{k_2}} \sec(k_2\pi/16), & k_1 = 0, \\ \tan(k_1\pi/16), & k_1 \neq 0, \end{cases} \quad (3.20)$$

and

$$\mathbf{W}_{cs/cc} = \begin{cases} \frac{\sqrt{8}}{\alpha_{k_1}} \sec(k_1\pi/16), & k_2 = 0, \\ \tan(k_2\pi/16), & k_2 \neq 0. \end{cases} \quad (3.21)$$

Hence, we can simply rewrite (3.18) as

$$\begin{aligned} \mathbf{C}_n = \mathbf{W}_{cc} \cdot * \left\{ \mathbf{W}_{cs/cc} \cdot * \left[(\mathbf{T}_c \mathbf{c}'_{1,n} \mathbf{T}'_c \mathbf{P}'_3) + \mathbf{W}_{ss/cs} \cdot * (\mathbf{P}_3 \mathbf{T}_c \mathbf{c}'_{3,n} \mathbf{T}'_c \mathbf{P}'_3) \right] \right. \\ \left. + \left[(\mathbf{T}_c \mathbf{c}_{0,n} \mathbf{T}'_c) + \mathbf{W}_{sc/cc} \cdot * (\mathbf{P}_3 \mathbf{T}_c \mathbf{c}'_{2,n} \mathbf{T}'_c) \right] \right\}. \end{aligned} \quad (3.22)$$

From (3.22), it is clear that the computation of fullband SBDCT requires four 2-D subband transforms $\mathbf{T}_c \mathbf{c}_{0,n} \mathbf{T}'_c$, $\mathbf{T}_c \mathbf{c}'_{1,n} \mathbf{T}'_c \mathbf{P}'_3$, $\mathbf{P}_3 \mathbf{T}_c \mathbf{c}'_{2,n} \mathbf{T}'_c$, and $\mathbf{P}_3 \mathbf{T}_c \mathbf{c}'_{3,n} \mathbf{T}'_c \mathbf{P}'_3$, where the permutations are defined by $\mathbf{P}_1$, $\mathbf{P}_2$, $\mathbf{P}_3$, and $\mathbf{P}_4$; and twiddle factors are defined by $\mathbf{W}_{cc}$, $\mathbf{W}_{cs/cc}$, $\mathbf{W}_{ss/cs}$, and $\mathbf{W}_{sc/cc}$.

Three observations can be made from the new twiddle factor matrices in (3.22). First, $\mathbf{W}_{sc/cc}$ and $\mathbf{W}_{cs/cc}$ each has eight '1's and $\mathbf{W}_{ss/cs}$ has seven '1's. In addition, $\mathbf{W}_{sc/cc}$ and $\mathbf{W}_{cs/cc}$ each has eight '0's and $\mathbf{W}_{ss/cs}$ has 15 '0's. No computational cost is incurred at these '1' and '0' elements. Second, $\mathbf{W}_{cs/cc}$ and $\mathbf{W}_{ss/sc}$ (except $k_1 = 0$, or $k_2 = 0$) have the same value for every column. Based on the fact that there are only 16 unique transform coefficients in $\mathbf{T}_c \mathbf{c}'_{1,n} \mathbf{T}'_c \mathbf{P}'_3$ and $\mathbf{T}_c \mathbf{c}'_{3,n} \mathbf{T}'_c$ each, some multiplications of $\mathbf{W}_{cs/cc} \cdot * (\mathbf{T}_c \mathbf{c}'_{1,n} \mathbf{T}'_c \mathbf{P}'_3)$ and $\mathbf{W}_{ss/sc} \cdot * (\mathbf{P}_3 \mathbf{T}_c \mathbf{c}'_{3,n} \mathbf{T}'_c \mathbf{P}'_3)$ can be avoided. Third, we have $\tan(l\pi/16) = 1/\tan((8-l)\pi/16)$ for $l < 8$, and this

relation becomes $\tan(6\pi/16) = 2 + \tan(2\pi/16)$ for $l = 2$. Thus, we can avoid the multiplication associated with $\tan(2\pi/16)$ and $\tan(6\pi/16)$ in (3.19) to (3.21), using one left shift and one addition.

Each twiddle factor matrix is of 8×8 , therefore, element-by-element multiplication with the twiddle factor matrix would require 64 multiplications. By taking into account the above observations mentioned earlier, the element-by-element multiplications of $\mathbf{W}_{cs/cc} \cdot *(\mathbf{T}_c \mathbf{c}'_{1,n} \mathbf{T}_c^t \mathbf{P}_3^t)$ and $\mathbf{W}_{ss/sc} \cdot *(\mathbf{P}_3 \mathbf{T}_c \mathbf{c}'_{3,n} \mathbf{T}_c^t \mathbf{P}_3^t)$ require 20 multiplications each instead of 64 multiplications. By considering the zero elements in $(\mathbf{T}_c \mathbf{c}'_{1,n} \mathbf{T}_c^t \mathbf{P}_3^t) + \mathbf{W}_{ss/cs} \cdot *(\mathbf{P}_3 \mathbf{T}_c \mathbf{c}'_{3,n} \mathbf{T}_c^t \mathbf{P}_3^t)$, the element-by-element multiplication of $\mathbf{W}'_{cs} / \mathbf{W}_{cc}$ and $(\mathbf{T}_c \mathbf{c}'_{1,n} \mathbf{T}_c^t \mathbf{P}_3^t) + \mathbf{W}_{ss/cs} \cdot *(\mathbf{P}_3 \mathbf{T}_c \mathbf{c}'_{3,n} \mathbf{T}_c^t \mathbf{P}_3^t)$ requires 42 multiplications. Due to the zero elements as well as symmetry of the transform coefficients and the twiddle factor matrices, the matrix additions of $(\mathbf{T}_c \mathbf{c}'_{1,n} \mathbf{T}_c^t \mathbf{P}_3^t) + \mathbf{W}_{ss/cs} \cdot *(\mathbf{P}_3 \mathbf{T}_c \mathbf{c}'_{3,n} \mathbf{T}_c^t \mathbf{P}_3^t)$ and $(\mathbf{T}_c \mathbf{c}'_{0,n} \mathbf{T}_c^t) + \mathbf{W}_{sc/cc} \cdot *(\mathbf{P}_3 \mathbf{T}_c \mathbf{c}'_{2,n} \mathbf{T}_c^t)$ require 24 additions each. Using the same observations, the matrix addition of $\mathbf{W}_{cs/cc} \cdot *[(\mathbf{T}_c \mathbf{c}'_{1,n} \mathbf{T}_c^t \mathbf{P}_3^t) + \mathbf{W}_{ss/cs} \cdot *(\mathbf{P}_3 \mathbf{T}_c \mathbf{c}'_{3,n} \mathbf{T}_c^t \mathbf{P}_3^t)]$ and $(\mathbf{T}_c \mathbf{c}'_{0,n} \mathbf{T}_c^t) + \mathbf{W}_{sc/cc} \cdot *(\mathbf{P}_3 \mathbf{T}_c \mathbf{c}'_{2,n} \mathbf{T}_c^t)$ are carried out with 48 additions.

3.2.1 Fast 4×4 DCT-II Algorithm

The 8×8 fullband SBDCT requires the computation of four 4×4 DCT-II, therefore, it is useful to derive a fast algorithm to compute the 4×4 DCT-II. Generally, there are two approaches (direct and indirect) to compute the 2-D DCT. The indirect approach involves computation of the 2-D DCT via other 2-D discrete orthogonal transforms (discrete Fourier transform [VN84, VN86], Walsh-

Hadamard transform [VKR88]). For the direct approach to compute the 2-D DCT-II, the separability property of the 2-D DCT-II transform kernel is used or the 2-D DCT-II kernel is directly computed using the 2-D input. The first method of the direct approach is commonly known as the row-column method. This approach first applies a 1-D DCT-II algorithm to the rows of the input and then transposes the immediate results, followed by applying the 1-D DCT-II algorithm to the columns of the immediate results. Therefore, the row-column method requires the computation of $2N$ N -point DCT-II. Using the fast 1-D DCT-II algorithm proposed by Loeffler *et al.* [LLM89], the row-column method requires a computational cost of (176, 464). The second method of the direct approach operates directly on an $N \times N$ input sequence to compute the 2-D DCT-II. Algorithms based on the direct 2-D DCT-II algorithms require significantly lesser computational cost as compared to those based on the row-column method. One major shortcoming of many algorithms based on the direct 2-D DCT-II computational structure is their high irregularity computational structure, which can lead to problems in implementation.

In [FW92], a fast algorithm for the 8×8 DCT-II is derived using an algebraic and computational theoretical approach. First, matrix factorization of the 8-point DCT-II matrix is converted to a sum of matrices corresponding to a set of irreducible polynomials. Using theorems on Kronecker products of matrices, an 8×8 DCT-II algorithm is derived from the results of matrix factorization of the 8-point DCT-II matrix. This derivation leads to a computation cost of (94, 454). Using polynomial transforms [DG90, PD96], the $N \times N$ DCT-II is mapped into a 2-D discrete Fourier transform (DFT) [RKH10], which is then used to evaluate a set of complex polynomials. Not only such approaches based on polynomial

transform possess complex computational structures, these approaches also require arithmetic of complex numbers. A class of 2-D DCT-II algorithms having moderate computational cost and regular computational structure was proposed in [WP89a, WP89b, CH91, WP91, BR00]. These algorithms are based on the vector-radix method, which decomposes the $N \times N$ DCT-II into a sum of four $N/2 \times N/2$ DCT-II, and the decomposition is repeated until 2×2 DCT-II is obtained. The algorithms based on vector-radix method require (144, 464).

Another class of 2-D DCT-II algorithms was proposed in [CYL91, CL91, CL92, CL93], which possesses regular computational structure. These algorithms are derived by replacing $\cos(\alpha)\cos(\beta)$ in the 2-D DCT-II kernel with $0.5(\cos(\alpha - \beta) + \cos(\alpha + \beta))$. These algorithms also use data reordering to map $N \times N$ DCT-II into N -point DCT-II with pre- and post-addition butterfly structures. Subsequently, the post-addition butterfly structure of the fast algorithm in [CL91] is further regularized by Huang *et al.* [HWH98, HW99]. Due to the mapping of $N \times N$ DCT-II into N N -point DCT-II, the required computational cost of these algorithms becomes directly proportional to the 1-D DCT-II algorithm, while some additions are required in the pre- and post-addition butterfly structures. For an 8×8 DCT-II, these fast algorithms have a computational cost of (96, 466).

For our derivation of the FSA, we adopted the algorithm proposed by Huang *et al.* [HWH98, HW99] due to its high regularity in its computational structure and low computational cost. In order to achieve optimal number of multiplications for the 4×4 DCT-II, the 4-point DCT-II should be optimal in terms of number of multiplications, as pointed out by Prado and Duhamel [PD96]. Therefore, we dedicate this subsection to the derivation of a fast 4-point DCT-II algorithm.

A fast 4×4 DCT-II algorithm with regular post addition structure was proposed in [HWH98, HW99], where Huang *et al.* mapped the $N \times N$ DCT-II into NN -point DCT-II based on the idea proposed by Cho and Lee [CL91]. For $N = 4$, the 4×4 DCT-II can be re-expressed as a sum of series of four 4-point DCT-II using the trigonometry identity $\cos(\alpha)\cos(\beta) = 0.5[\cos(\alpha - \beta) + \cos(\alpha + \beta)]$, which is

$$C(\mathbf{k}, \mathbf{n}) = \begin{cases} \frac{1}{2} \sum_{p_1=0}^1 \sum_{p_2=0}^3 (\bar{c}(\mathbf{p}, \mathbf{n}) + \bar{c}(p_1 + 2, p_2, \mathbf{n})) \\ \quad \cdot \left(W_c^{(4p_2+3)[(4p_1+1)k_1+k_2]/8} + W_c^{(4p_2+3)[(4p_1+1)k_1-k_2]/8} \right), & k_1 = 0, 2, \\ \frac{1}{2} \sum_{p_1=0}^1 \sum_{p_2=0}^3 (\bar{c}(\mathbf{p}, \mathbf{n}) - \bar{c}(p_1 + 2, p_2, \mathbf{n})) \\ \quad \cdot \left(W_c^{(4p_2+3)[(4p_1+1)k_1+k_2]/8} + W_c^{(4p_2+3)[(4p_1+1)k_1-k_2]/8} \right), & k_1 = 1, 3, \end{cases} \quad (3.23)$$

where $\bar{c}(\mathbf{p}, \mathbf{n}) = \tilde{c}(\text{mod}((4p_2 + 3)p_1 + p_2, 4), p_2, \mathbf{n})$, $k_2 = 0, 1, 2, 3$, and $\tilde{c}(\mathbf{p}, \mathbf{n})$ is given by

$$\tilde{c}(\mathbf{p}, \mathbf{n}) = \begin{cases} c(2p_1 + 1, 2p_2 + 1, \mathbf{n}), & 0 < p_1, p_2 \leq 1, \\ c(2p_1 + 1, 6 - 2p_2, \mathbf{n}), & 0 < p_1 \leq 1, 2 \leq p_2 \leq 3, \\ c(6 - 2p_1, 2p_2 + 1, \mathbf{n}), & 2 \leq p_1 \leq 3, 0 \leq p_2 \leq 1, \\ c(6 - 2p_1, 6 - 2p_2, \mathbf{n}), & 2 \leq p_1 \leq 3, 2 \leq p_2 \leq 3. \end{cases} \quad (3.24)$$

Further decomposition of (3.23) using $a(\mathbf{p}, \mathbf{n}) = \bar{c}(p_1, p_2, \mathbf{n}) + \bar{c}(p_1 + 2, p_2, \mathbf{n})$ and

$b(\mathbf{p}, \mathbf{n}) = \bar{c}(p_1, p_2, \mathbf{n}) - \bar{c}(p_1 + 2, p_2, \mathbf{n})$ yields

$$C(\mathbf{k}, \mathbf{n}) = \begin{cases} \sum_{p_2=0}^3 [a(0, p_2, \mathbf{n}) + a(1, p_2, \mathbf{n})] W_c^{(4p_2+3)k_2/8}, & k_1 = 0, \\ \frac{1}{2} \sum_{p_2=0}^3 \left[b(0, p_2, \mathbf{n}) \left(W_c^{(4p_2+3)(1+k_2)/8} + W_c^{(4p_2+3)(1-k_2)/8} \right) \right. \\ \quad \left. - b(1, p_2, \mathbf{n}) \left(W_c^{(4p_2+3)(3+k_2)/8} + W_c^{(4p_2+3)(3-k_2)/8} \right) \right], & k_1 = 1, \\ \frac{1}{2} \sum_{p_2=0}^3 (a(0, p_2, \mathbf{n}) - a(1, p_2, \mathbf{n})) \left(W_c^{(4p_2+3)(2+k_2)/8} + W_c^{(4p_2+3)(2-k_2)/8} \right), & k_1 = 2, \\ \frac{1}{2} \sum_{p_2=0}^3 \left[b(0, p_2, \mathbf{n}) \left(W_c^{(4p_2+3)(3+k_2)/8} + W_c^{(4p_2+3)(3-k_2)/8} \right) \right. \\ \quad \left. + b(1, p_2, \mathbf{n}) \left(W_c^{(4p_2+3)(1+k_2)/8} + W_c^{(4p_2+3)(1-k_2)/8} \right) \right], & k_1 = 3. \end{cases} \quad (3.25)$$

By permutation, we have the following:

$$\begin{aligned} C(k_1, n_1) &= \sum_{p_1=0}^3 c(p_1, n_1) W_c^{(4p_1+3)k_1/8} \\ &= \sum_{p_1=0}^3 \hat{c}(p_1, n_1) W_c^{(2p_1+1)k_1/8}, \end{aligned} \quad (3.26)$$

where

$$\hat{c}(p_1, n_1) = \begin{cases} c\left(\frac{p_1-1}{2}, n_1\right), & \text{for even } p_1, \\ c\left(3-\frac{p_1}{2}, n_1\right), & \text{for odd } p_1. \end{cases} \quad (3.27)$$

Hence, (3.25) to (3.27) reveal that the 4×4 DCT-II can be computed from summation of four 4-point DCT-II.

In the following, we consider the 4-point fast DCT-II algorithm [TGW05b].

Since $W_c^{(8k_1-ak_1)/8} = (-1)^{k_1} W_c^{ak_1/8}$, we can rewrite (3.1) as

$$C(k_1, n_1) = \left[c(0, n_1) + (-1)^{k_1} c(3, n_1) \right] W_c^{k_1/8} + \left[c(1, n_1) + (-1)^{k_1} c(2, n_1) \right] W_c^{3k_1/8}, \quad (3.28)$$

For odd values of k_1 , (3.28) becomes

$$\begin{aligned} C(1, n_1) &= (c(0, n_1) - s(3, n_1))W_c^{1/8} + (c(1, n_1) - c(2, n_1))W_c^{3/8}, \\ C(3, n_1) &= (c(0, n_1) - s(3, n_1))W_c^{3/8} - (c(1, n_1) - c(2, n_1))W_c^{1/8}, \end{aligned} \quad (3.29)$$

which is a butterfly pair. Following the method in [LLM89], the butterfly pair in (3.29) is computed using

$$\begin{aligned} C(1, n_1) &= (W_c^{3/8} - W_c^{1/8})(c(1, n_1) - c(2, n_1)) \\ &\quad + W_c^{1/8}(c(1, n_1) - c(2, n_1) + c(0, n_1) - c(3, n_1)), \\ C(3, n_1) &= (W_c^{3/8} + W_c^{1/8})(c(0, n_1) - c(3, n_1)) \\ &\quad + W_c^{1/8}(c(1, n_1) - c(2, n_1) + c(0, n_1) - c(3, n_1)), \end{aligned} \quad (3.30)$$

which has a computational cost of (3, 3). This butterfly pair is encapsulated in the *cosB* block. Let $o_{\cos B}(j)$ and $i_{\cos B}(j)$, where $j = 0, 1$, denote the output and input of the *cosB* block, respectively. *cosB* is defined as

$$\begin{aligned} o_{\cos B}(0) &= (W_s^{1/8} - W_c^{1/8})i_{\cos B}(0) + W_c^{1/8}(i_{\cos B}(0) + i_{\cos B}(1)), \\ o_{\cos B}(1) &= (W_s^{1/8} + W_c^{1/8})i_{\cos B}(1) - W_c^{1/8}(i_{\cos B}(0) + i_{\cos B}(1)). \end{aligned} \quad (3.31)$$

For even values of k_1 , (3.28) becomes

$$\begin{aligned} C(0, n_1) &= (c(1, n_1) + c(2, n_1)) + (c(0, n_1) + c(3, n_1)), \\ C(2, n_1) &= \frac{1}{\sqrt{2}}[(c(1, n_1) + c(2, n_1)) - (c(0, n_1) + c(3, n_1))]. \end{aligned} \quad (3.32)$$

Equation (3.32) is a butterfly pair with a computational cost of (1, 3). By combining (3.30) and (3.32), the computational cost for the 4-point DCT-II becomes (4, 9). The computational structure of the proposed fast 4-point DCT-II algorithm is illustrated in Fig. 3.2. By substituting (3.30) and (3.32) into (3.25), a fast 4×4 DCT-II algorithm based on the Huang's approach [HWH98, HW99] is obtained. The computational structure of the proposed fast 4×4 DCT-II algorithm is shown in Fig. 3.3. The 16 multiplications at the right of the flow diagram can be performed by right shifting the output, and the computational cost of the 4×4 fast

DCT-II algorithm is found to be (16, 74). Therefore, the 4-point and 4×4 DCT-II algorithms achieve the theoretical lower bound of multiplications given by $2^{n+1} - n - 2$ [HB85, DH87, WF92] and $2^n(2^{n+1} - n - 2)$ [WF92], respectively, where $N = 2^n$.

Fig. 3.2 Computational structure of 4-point DCT-II. The computational cost of this structure is (4, 9).

Fig. 3.3 Computational structure of 4×4 DCT-II. The computational cost of this structure is (16, 74) and 9 right shifts.

3.2.2 Computational Cost Analysis of Fast Subband Discrete Cosine Transform and Approximations of DCT-II

The computational cost of the FSA is determined by the subband decomposition to obtain four subbands, four 2-D subband transforms and a weighting network, as shown in Fig. 3.1. In the following, we shall derive the computational cost of various methods to compute SBDCT.

From (3.8), it is observed that each column of $\mathbf{T}_c$ and $\mathbf{T}_s$ contains one '0' and '1', hence, computing the 4-point subband DCT and subband DST require a computational cost of ($4 \times 6 = 24$, $3 \times 7 = 21$). Using row-column decomposition, the computational cost of each 2-D subband transform arrives to ($24 \times 4 + 24 \times 8 = 288$, $21 \times 4 + 21 \times 8 = 252$). In the weighting network, $3 \times 64 = 192$ additions are required to add the four 2-D subband transforms, and $4 \times 64 = 256$ multiplications are required for the multiplication of the twiddle factors and the four 2-D subband transforms. Thus, computational cost of the fullband SBDCT becomes

$$\begin{aligned}
 c_m &= \text{Multiplication required by Subband Transform} \\
 &\quad + \text{Multiplication required by Twiddle Factors} \\
 &= (24 \times 4 + 24 \times 8) \times 4 + (64 \times 4) \\
 &= 1408, \\
 c_a &= \text{Additions required by Subband Decomposition} \\
 &\quad + \text{Additions required by Subband Transform} \\
 &\quad + \text{Additions required by adding Subband Transform} \\
 &= (3 \times 16) \times 4 + (21 \times 4 + 21 \times 8) \times 4 + 192 \\
 &= 1392.
 \end{aligned} \tag{3.33}$$

Considering the odd and even symmetries of the 4-point subband DCT and subband DST, we only compute four transform coefficients instead of eight for the 4-point subband DCT and subband DST. This observation reduces the

computational cost of the 4-point subband DCT and subband DST to ($4 \times 3 = 12$, $3 \times 4 = 12$). With this in mind, the computational cost of each 2-D subband transform becomes ($12 \times 4 + 12 \times 8 = 144$, $12 \times 4 + 12 \times 8 = 144$). In addition, each 2-D subband transform contains 15 zero elements, therefore, the number of multiplications required by the twiddle factors reduces to $4 \times (64 - 15) = 196$. The computational cost of the fullband SBDCT is then reduced to

$$\begin{aligned}
c_m &= \text{Multiplications required by Subband Transform} \\
&\quad + \text{Multiplications required by Twiddle Factors} \\
&= (12 \times 4 + 12 \times 8) \times 4 + [(64 - 15) \times 4] \\
&= 772, \\
c_a &= \text{Additions required by Subband Decomposition} \\
&\quad + \text{Additions required by Subband Transform} \\
&\quad + \text{Additions required by adding Subband Transform} \\
&= (3 \times 16) \times 4 + (12 \times 4 + 12 \times 8) \times 4 + 192 \\
&= 960.
\end{aligned} \tag{3.34}$$

In [TGW05b, TG11], we also proposed a computational structure for subband DCT and subband DST. Their computational structures are shown in Fig. 3.4 and 3.5, respectively. Let $o_{sinB}(j)$ and $i_{sinB}(j)$, where $j = 0, 1$, denote the output and input of $sinB$, respectively, and the definition of $sinB$ in Fig. 3.5 is

$$\begin{aligned}
o_{sinB}(0) &= (W_c^{3/8} - W_c^{1/8})i_{sinB}(0) + W_c^{1/8}(i_{sinB}(0) + i_{sinB}(1)), \\
o_{sinB}(1) &= -(W_c^{3/8} + W_c^{1/8})i_{sinB}(1) + W_c^{1/8}(i_{sinB}(0) + i_{sinB}(1)).
\end{aligned} \tag{3.35}$$

Combining the structures in Fig. 3.4 and 3.5, a FSA is obtained [TGW05b, TG11]. For the 8×8 SBDCT, this computational structure requires (308, 552), and achieves significant savings in multiplications as compared to the computational cost of (432, 544) required by the FSA in [JMM96].

Fig. 3.4 Signal flow diagram of N -point subband DCT. For a 4-point input, $C(0,4,n_1)$ is 0 and this computational structure requires 3 multiplications and 9 additions.

Fig. 3.5 Signal flow diagram of N -point subband DST. For a 4-point input, $C(1,0,n_1)$ is 0 and this computational structure requires 4 multiplications and 9 additions.

Based on (3.22), the computational cost of the proposed FSA is determined by the subband decomposition of the input sequence, four 4×4 DCT-II, and four twiddle factor matrices. The step-by-step derivation of computational cost of the proposed FSA is presented in Table 3.2. The quantization of the SBDCT coefficient can be included in (3.22) with an additional cost of four multiplications and four arithmetic shifts. Let $\hat{\mathbf{C}}_n$ and $\mathbf{Q}$ denote the quantized SBDCT

coefficients $\mathbf{C}_n$ and the quantization matrix, respectively. The quantized SBDCT coefficient are computed as

$$\begin{aligned} \hat{\mathbf{C}}_n = \text{round} & \left(\hat{\mathbf{W}}_{cc} \cdot * \left\{ \mathbf{W}_{cs/cc} \cdot * \left[\left(\mathbf{T}_c \mathbf{c}'_{1,n} \mathbf{T}_c' \mathbf{P}_3' \right) + \mathbf{W}_{ss/cs} \cdot * \left(\mathbf{P}_3 \mathbf{T}_c \mathbf{c}'_{3,n} \mathbf{T}_c' \mathbf{P}_3' \right) \right] \right. \right. \\ & \left. \left. + \left[\left(\mathbf{T}_c \mathbf{c}_{0,n} \mathbf{T}_c' \right) + \mathbf{W}_{sc/cc} \cdot * \left(\mathbf{P}_3 \mathbf{T}_c \mathbf{c}'_{2,n} \mathbf{T}_c' \right) \right] \right\} \right), \end{aligned} \quad (3.36)$$

where $\hat{\mathbf{W}}_{cc} = \mathbf{W}_{cc} / \mathbf{Q}$, and the $\text{round}(\cdot)$ operator produces the nearest integer value. Finally, we have the computational cost of the proposed FSA with and without quantization as (208, 536) and (204, 536), respectively. As compared to existing FSAs, the proposed FSA achieves up to 52 % reduction of multiplications.

The energy content of most images is found to be predominantly concentrated in the low-low subband, which is the $\mathbf{c}_{0,n}$ for the case of the SBDCT. Hence, an approximation of the DCT-II based on the SBDCT can be obtained by selectively computing the four 2-D subband transforms in (3.12), where the simplest approximation is $\mathbf{C}_n \approx \mathbf{W}_{cc} \cdot * (\mathbf{T}_c \mathbf{c}_{0,n} \mathbf{T}_c')$ [JMM96]. Jung *et al.* [JMM96] proposed further decomposing $\mathbf{c}_{0,n}$, and estimating $\mathbf{c}_{0,n}$ using its dominant subband to achieve higher saving of computational cost at the expense of image quality. Since the objective of this thesis is to achieve high computational cost saving while maintaining image quality, further decomposition of $\mathbf{c}_{0,n}$ is avoided.

Assuming that the n th $N \times N$ sub-image does not have any energy content in one or more subbands, we arrive at seven possibilities of approximating DCT-II based on the SBDCT, which are referred as ApD1 to ApD7. The computational cost (with and without quantization matrix) of the seven approximations of DCT-II based on SBDCT is summarized in Table 3.3. Considering a single level of decomposition, the approximation of DCT-II in [JMM96, HH04] is essentially the same as ApD1.

Table 3.2 Derivation of computational cost of proposed FSA.

s/n	Operation	Mult.	Addition	Shift
1	Decompose $\mathbf{c}_n$ to $\mathbf{c}_{0,n}$, $\mathbf{c}_{1,n}$, $\mathbf{c}_{2,n}$, and $\mathbf{c}_{3,n}$	0	128	0
2	$\mathbf{T}'_c \mathbf{c}_{0,n} (\mathbf{T}'_c)^t$	16	74	9
3	$\mathbf{T}'_c \mathbf{c}'_{1,n} (\mathbf{T}'_c)^t$	16	74	9
4	$\mathbf{T}'_c \mathbf{c}'_{2,n} (\mathbf{T}'_c)^t$	16	74	9
5	$\mathbf{T}'_c \mathbf{c}'_{3,n} (\mathbf{T}'_c)^t$	16	74	9
6	Multiplication of $\mathbf{P}_3$ with 3 rd , 4 th and 5 th operations	0	0	0
7	Element-by-element multiplication of $\mathbf{W}_{ss/cs}$ and $\mathbf{P}_3 \mathbf{T}_c \mathbf{c}'_{3,n} \mathbf{T}'_c \mathbf{P}_3^t$	20	4	4
8	Element-by-element multiplication of $\mathbf{W}_{sc/cc}$ and $\mathbf{P}_3 \mathbf{T}_c \mathbf{c}'_{2,n} \mathbf{T}'_c$	20	4	4
9	Addition of $\mathbf{T}_c \mathbf{c}'_{1,n} \mathbf{T}'_c \mathbf{P}_3^t$ and $\mathbf{W}_{ss/cs} \cdot * (\mathbf{P}_3 \mathbf{T}_c \mathbf{c}'_{3,n} \mathbf{T}'_c \mathbf{P}_3^t)$	0	24	0
10	Addition of $\mathbf{T}_c \mathbf{c}_{0,n} \mathbf{T}'_c$ and $\mathbf{W}_{sc/cc} \cdot * (\mathbf{P}_3 \mathbf{T}_c \mathbf{c}'_{2,n} \mathbf{T}'_c)$	0	24	0
11	Element-by-element multiplication of $\mathbf{W}_{cs/cc}$ and 9 th operation	40	8	8
12	Addition of 10 th and 11 th operations	0	48	0
13	Element-by-element multiplication of $\mathbf{W}_{cc}$ and 12 th operation	60	0	4
Total		204	536	56

Table 3.3 Computational cost of approximation of DCT-II using FSA.

Type (Subband)	Approximation of DCT-II (Direct Form)	Approximation of DCT-II (using FSA)	Computational Cost (c_m, c_a)	
			FSA	FSA with Q
$\mathbf{C}_n^{\text{ApD1}}$ (LL)	$\mathbf{W}_{cc} \cdot * (\mathbf{T}_{c_{0,n}} \mathbf{T}_c')$	$\mathbf{W}_{cc} \cdot * (\mathbf{T}_{c_{0,n}} \mathbf{T}_c')$	(64, 126)	(65, 126)
$\mathbf{C}_n^{\text{ApD2}}$ (LL + LH)	$\mathbf{W}_{cc} \cdot * (\mathbf{T}_{c_{0,n}} \mathbf{T}_c') + \mathbf{W}_{cs} \cdot * (\mathbf{T}_{c_{1,n}} \mathbf{T}_c' \mathbf{P}_3')$	$\mathbf{W}_{cc} \cdot * [(\mathbf{T}_{c_{0,n}} \mathbf{T}_c') + (\mathbf{W}_{cs} \cdot / \mathbf{W}_{cc}) \cdot * (\mathbf{T}_{c_{1,n}} \mathbf{T}_c' \mathbf{P}_3')]$	(106, 240)	(108, 240)
$\mathbf{C}_n^{\text{ApD3}}$ (LL + HL)	$\mathbf{W}_{cc} \cdot * (\mathbf{T}_{c_{0,n}} \mathbf{T}_c') + \mathbf{W}_{sc} \cdot * (\mathbf{P}_3 \mathbf{T}_{c_{2,n}} \mathbf{T}_c')$	$\mathbf{W}_{cc} \cdot * [(\mathbf{T}_{c_{0,n}} \mathbf{T}_c') + (\mathbf{W}_{sc} \cdot / \mathbf{W}_{cc}) \cdot * (\mathbf{P}_3 \mathbf{T}_{c_{2,n}} \mathbf{T}_c')]$	(106, 240)	(108, 240)
$\mathbf{C}_n^{\text{ApD4}}$ (LL + HH)	$\mathbf{W}_{cc} \cdot * (\mathbf{T}_{c_{0,n}} \mathbf{T}_c') + \mathbf{W}_{ss} \cdot * (\mathbf{P}_3 \mathbf{T}_{c_{3,n}} \mathbf{T}_c' \mathbf{P}_3')$	$\mathbf{W}_{cc} \cdot * [(\mathbf{T}_{c_{0,n}} \mathbf{T}_c') + (\mathbf{W}_{ss} \cdot / \mathbf{W}_{cc}) \cdot * (\mathbf{P}_3 \mathbf{T}_{c_{3,n}} \mathbf{T}_c' \mathbf{P}_3')]$	(134, 248)	(136, 248)
$\mathbf{C}_n^{\text{ApD5}}$ (LL + LH + HL)	$\mathbf{W}_{cc} \cdot * (\mathbf{T}_{c_{0,n}} \mathbf{T}_c') + \mathbf{W}_{cs} \cdot * (\mathbf{T}_{c_{1,n}} \mathbf{T}_c' \mathbf{P}_3') + \mathbf{W}_{sc} \cdot * (\mathbf{P}_3 \mathbf{T}_{c_{2,n}} \mathbf{T}_c')$	$\mathbf{W}_{cc} \cdot * [(\mathbf{T}_{c_{0,n}} \mathbf{T}_c') + (\mathbf{W}_{cs} \cdot / \mathbf{W}_{cc}) \cdot * (\mathbf{T}_{c_{1,n}} \mathbf{T}_c' \mathbf{P}_3') + (\mathbf{W}_{sc} \cdot / \mathbf{W}_{cc}) \cdot * (\mathbf{P}_3 \mathbf{T}_{c_{2,n}} \mathbf{T}_c')]$	(148, 392)	(151, 392)
$\mathbf{C}_n^{\text{ApD6}}$ (LL + LH + HH)	$\mathbf{W}_{cc} \cdot * (\mathbf{T}_{c_{0,n}} \mathbf{T}_c') + \mathbf{W}_{cs} \cdot * (\mathbf{T}_{c_{1,n}} \mathbf{T}_c' \mathbf{P}_3') + \mathbf{W}_{ss} \cdot * (\mathbf{P}_3 \mathbf{T}_{c_{3,n}} \mathbf{T}_c' \mathbf{P}_3')$	$\mathbf{W}_{cc} \cdot * [(\mathbf{T}_{c_{0,n}} \mathbf{T}_c') + (\mathbf{W}_{cs} \cdot / \mathbf{W}_{cc}) \cdot * (\mathbf{T}_{c_{1,n}} \mathbf{T}_c' \mathbf{P}_3') + (\mathbf{W}_{ss} \cdot / \mathbf{W}_{cc}) \cdot * (\mathbf{P}_3 \mathbf{T}_{c_{3,n}} \mathbf{T}_c' \mathbf{P}_3')]$	(148, 392)	(157, 392)
$\mathbf{C}_n^{\text{ApD7}}$ (LL + HL + HH)	$\mathbf{W}_{cc} \cdot * (\mathbf{T}_{c_{0,n}} \mathbf{T}_c') + \mathbf{W}_{sc} \cdot * (\mathbf{P}_3 \mathbf{T}_{c_{2,n}} \mathbf{T}_c') + \mathbf{W}_{ss} \cdot * (\mathbf{P}_3 \mathbf{T}_{c_{3,n}} \mathbf{T}_c' \mathbf{P}_3')$	$\mathbf{W}_{cc} \cdot * [(\mathbf{T}_{c_{0,n}} \mathbf{T}_c') + (\mathbf{W}_{sc} \cdot / \mathbf{W}_{cc}) \cdot * (\mathbf{T}_{c_{2,n}} \mathbf{T}_c' \mathbf{P}_3') + (\mathbf{W}_{ss} \cdot / \mathbf{W}_{cc}) \cdot * (\mathbf{P}_3 \mathbf{T}_{c_{3,n}} \mathbf{T}_c' \mathbf{P}_3')]$	(148, 392)	(157, 392)

3.3 Adaptive Subband Discrete Cosine Transform

Based on the assumption that most of the energy of $c(p_1, n_1)$ lies in $c(0, q_1, n_1)$, a good approximation of the 1-D DCT-II can be obtained by just computing the first term on the RHS of (3.2). Let $\tilde{C}(k_1, n_1)$ denote this approximation of the 1-D DCT-II. From (3.2) and (3.9), the matrix representation of $\tilde{C}(k_1, n_1)$ is expressed as

$$\tilde{\mathbf{C}}_{n_1}^{\text{1D}} = \mathbf{D} \mathbf{T}_c \mathbf{D}_c \mathbf{P}_2 (\mathbf{I}_4 \otimes \mathbf{P}_1) \mathbf{c}_{n_1}, \quad (3.37)$$

where $\mathbf{c}_{n_1}$ is a n_1 th N -point sequence. The MSE between $\tilde{\mathbf{C}}_{n_1}^{\text{1D}}$ and the DCT-II is computed as [JT01]

$$\varepsilon = \frac{1}{N} \text{trace}(\mathbf{D}_T \mathbf{R} \mathbf{D}_T^t), \quad (3.38)$$

where $\text{trace}(\cdot)$ returns the trace of a matrix; $\mathbf{R}$ is the covariance matrix of a stationary Markov-I signal, where the elements of the correlation matrix $\mathbf{R}$ is characterized by

$$r(\mathbf{m}) = \rho^{|m_1 - m_2|}, \quad m_1, m_2 = 0, 1, \dots, N-1, \quad (3.39)$$

ρ is the inter-element correlation coefficient; $\mathbf{m} = \{m_1, m_2\}$; $\mathbf{D}_T$ is computed as

$$\mathbf{D}_T = \mathbf{T} - \mathbf{D} \mathbf{T}_c \mathbf{D}_c \mathbf{P}_2 (\mathbf{I}_4 \otimes \mathbf{P}_1), \quad (3.40)$$

and $\mathbf{T}$ is the basis vector of DCT-II. Using (3.38), the MSE between $\tilde{\mathbf{C}}_{n_1}^{\text{1D}}$ and DCT-II is plotted in Fig. 3.6. It is also observed that MSE is independent of N , and MSE reduces linearly with increasing ρ . From Fig. 3.6, it is obvious that (3.37) should only be applied when $\mathbf{c}_{n_1}$ is highly correlated.

Fig. 3.6 MSE between $\tilde{\mathbf{C}}_{n_1}^{1D}$ and DCT-II.

Adapting the aliased component analysis on SBDCT [HH04], we analyze the aliased components in $\tilde{\mathbf{C}}_{n_1}^{1D}$. To further understand the MSE between $\tilde{\mathbf{C}}_{n_1}^{1D}$ and DCT-II, $\tilde{\mathbf{C}}_{n_1}^{1D}$ is re-expressed as

$$\begin{aligned}
\tilde{C}(k_1, n_1) &= 2\alpha_{k_1} W_c^{k_1/2N} C(0, k_1, n_1) \\
&= 2\alpha_{k_1} W_c^{k_1/2N} \sum_{q_1=0}^{N'-1} c(0, q_1, n_1) W_c^{(2q_1+1)k_1/N} \\
&= \alpha_{k_1} W_c^{k_1/2N} \sum_{q_1=0}^{N'-1} c(2q_1, n_1) \left(W_c^{(4q_1+1)k_1/2N} W_c^{k_1/2N} - W_s^{(4q_1+1)k_1/2N} W_s^{k_1/2N} \right) \\
&\quad + \alpha_{k_1} W_c^{k_1/2N} \sum_{q_1=0}^{N'-1} c(2q_1+1, n_1) \left(W_c^{(4q_1+3)k_1/2N} W_c^{k_1/2N} + W_s^{(4q_1+3)k_1/2N} W_s^{k_1/2N} \right) \\
&= \left(W_c^{k_1/2N} \right)^2 C(k_1, n_1) - \alpha_{k_1} W_c^{k_1/2N} W_s^{k_1/2N} \sum_{q_1=0}^{N'-1} c(2q_1, n_1) W_s^{(4q_1+1)k_1/2N} \\
&\quad + \alpha_{k_1} W_c^{k_1/2N} W_s^{k_1/2N} \sum_{q_1=0}^{N'-1} c(2q_1+1, n_1) W_s^{(4q_1+3)k_1/2N} \\
&= \alpha_{k_1} W_c^{k_1/2N} \left(\frac{W_c^{k_1/2N}}{\alpha_{k_1}} C(k_1, n_1) - W_s^{k_1/2N} C(k'_1, n_1) \right),
\end{aligned} \tag{3.41}$$

using the relation $W_c^{(4k_1+1)(k_2-N)/2N} = W_s^{(4k_1+1)k_2/2N}$, $W_c^{(4k_1+3)(k_2-N)/2N} = -W_s^{(4k_1+3)k_2/2N}$, and $k'_1 = k_1 - N$. The first term on the RHS of (3.41) indicates that $C(k_1, n_1)$ is diminished by a factor of $\alpha_{k_1} (W_c^{k_1/2N})^2$. This diminishing factor can be viewed as a linear distortion of the 1-D SBDCT [HH04]. The second term on the RHS of (3.41) indicates that aliasing in $C(k_1, n_1)$ is caused by the transform coefficients in $C(k'_1, n_1)$. By normalizing (3.41) by $\alpha_{k_1} (W_c^{k_1/2N})^2$, we have

$$\frac{\tilde{C}(k_1, n_1)}{\alpha_{k_1} (W_c^{k_1/2N})^2} = \frac{1}{\alpha_{k_1}} C(k_1, n_1) - \frac{W_s^{k_1/2N}}{W_c^{k_1/2N}} C(k'_1, n_1), \quad (3.42)$$

where the $C(k_1, n_1)$ and $C(k'_1, n_1)$ terms are the 1-D DCT-II and the normalized aliased component, respectively.

To approximate the 2-D DCT-II based on 2-D SBDCT, we have

$$\begin{aligned} C^{\text{ApD1}}(\mathbf{k}, \mathbf{n}) &= 4\alpha_{k_1}\alpha_{k_2} W_c^{k_1/2N} W_c^{k_2/2N} C(0, \mathbf{k}, \mathbf{n}) \\ &= \alpha_{k_1}\alpha_{k_2} (W_c^{k_1/2N})^2 (W_c^{k_2/2N})^2 \left(\frac{C(\mathbf{k}, \mathbf{n})}{\alpha_{k_1}\alpha_{k_2}} - \frac{W_s^{k_2/2N}}{W_c^{k_2/2N}} C(k_1, k'_2, \mathbf{n}) \right. \\ &\quad \left. - \frac{W_s^{k_1/2N}}{W_c^{k_1/2N}} C(k'_1, k_2, \mathbf{n}) + \frac{W_s^{k_1/2N} W_s^{k_2/2N}}{W_c^{k_1/2N} W_c^{k_2/2N}} C(\mathbf{k}', \mathbf{n}) \right), \end{aligned} \quad (3.43)$$

where $k'_2 = k_2 - N$, and $\mathbf{k}' = k'_1, k'_2$. Similar to the 1-D case, the $C(\mathbf{k}, \mathbf{n})$ term on the RHS of (3.43) is diminished by a factor of $\alpha_{k_1}\alpha_{k_2} (W_c^{k_1/2N})^2 (W_c^{k_2/2N})^2$. The aliased components of the 2-D SBDCT after normalization are

$$\begin{aligned} \frac{C^{\text{ApD1}}(\mathbf{k}, \mathbf{n})}{\alpha_{k_1}\alpha_{k_2} (W_c^{k_1/2N})^2 (W_c^{k_2/2N})^2} &= \frac{4}{W_c^{k_1/2N} W_c^{k_2/2N}} C(0, \mathbf{k}, \mathbf{n}) \\ &= \frac{1}{\alpha_{k_1}\alpha_{k_2}} C(\mathbf{k}, \mathbf{n}) - \frac{W_c^{k_1/2N} W_s^{k_2/2N}}{W_c^{k_2/2N}} C(k_1, k'_2, \mathbf{n}) \\ &\quad - \frac{W_s^{k_1/2N}}{W_c^{k_1/2N}} C(k'_1, k_2, \mathbf{n}) + \frac{W_s^{k_1/2N} W_s^{k_2/2N}}{W_c^{k_1/2N} W_c^{k_2/2N}} C(\mathbf{k}', \mathbf{n}), \end{aligned} \quad (3.44)$$

and it is clear that aliasing in $C^{\text{ApD1}}(\mathbf{k}, \mathbf{n})$ is contributed by $C(k_1, k'_2, \mathbf{n})$, $C(k'_1, k_2, \mathbf{n})$, and $C(\mathbf{k}', \mathbf{n})$.

Assuming that $C(\mathbf{k}, \mathbf{n})$ is a flat spectrum with DCT coefficients having a magnitude of one, the contribution of the aliased components in (3.44) is computed and summarized in Table 3.4 and 3.5. This assumption neglects the differences of the magnitude of the various DCT coefficients, and aims to determine the worst case of aliasing if one compares the aliasing from one DCT coefficient to another. From Table 3.4, it is observed that the highest amount of aliasing of $C(k_1, k'_2, \mathbf{n})$ and $C(k'_1, k_2, \mathbf{n})$ is found in its $(N-1)$ th column and row, respectively. If the aliased component $C(k_1, k'_2, \mathbf{n})$ and $C(k'_1, k_2, \mathbf{n})$ are arranged into $N \times N$ matrices, it would be found that the matrix formed by aliased components $C(k_1, k'_2, \mathbf{n})$ and $C(k'_1, k_2, \mathbf{n})$ are transposed of each other. This is expected since these aliased components are scaled by $W_s^{k_2/2N} / W_c^{k_2/2N}$ and $W_s^{k_1/2N} / W_c^{k_1/2N}$, respectively. From Table 3.5, the largest contribution of the aliased component $C(\mathbf{k}', \mathbf{n})$ is found at $k'_1, k'_2 = \{N-1, N-1\}$. Also, it is observed that aliasing in $\mathbf{C}_n^{\text{ApD1}}$ is increased when N increases. This clearly indicates that aliasing increases when a larger block size is used.

For the 2-D SBDCT, four subband transforms are considered and the aliased components of the three other transforms (the first transform is defined in $\mathbf{C}_n^{\text{ApD1}}$) are

$$\begin{aligned} \frac{4}{W_c^{k_1/2N} W_s^{k_2/2N}} C(1, \mathbf{k}, \mathbf{n}) &= \frac{1}{\alpha_{k_1} \alpha_{k_2}} C(\mathbf{k}, \mathbf{n}) + \frac{W_c^{k_2/2N}}{W_s^{k_2/2N}} C(k_1, k'_2, \mathbf{n}) \\ &\quad - \frac{W_s^{k_1/2N}}{W_c^{k_1/2N}} C(k'_1, k_2, \mathbf{n}) - \frac{W_s^{k_1/2N} W_c^{k_2/2N}}{W_c^{k_1/2N} W_s^{k_2/2N}} C(\mathbf{k}', \mathbf{n}), \end{aligned} \quad (3.45)$$

Table 3.4 Normalized aliased components of $\mathbf{C}_n^{\text{ApD1}}$.

Alias at $C(k_1, k'_2, \mathbf{n})$				Alias at $C(k'_1, k_2, \mathbf{n})$			
k'_2	$k_1 = 0, 1, \dots, N-1$			k'_1	$k_2 = 0, 1, \dots, N-1$		
	$N=2$	$N=4$	$N=8$		$N=2$	$N=4$	$N=8$
0	0.000	0.000	0.000	0	0.000	0.000	0.000
1	-1.000	-0.414	-0.199	1	-1.000	-0.414	-0.199
2		-1.000	0.414	2		-1.000	0.414
3		-2.414	-0.668	3		-2.414	-0.668
4			-1.000	4			-1.000
5			-1.497	5			-1.497
6			-2.414	6			-2.414
7			-5.027	7			-5.027

$$\begin{aligned}
 \frac{4}{W_s^{k_1/2N} W_c^{k_2/2N}} C(2, \mathbf{k}, \mathbf{n}) &= \frac{1}{\alpha_{k_1} \alpha_{k_2}} C(\mathbf{k}, \mathbf{n}) - \frac{W_s^{k_2/2N}}{W_c^{k_2/2N}} C(k_1, k'_2, \mathbf{n}) \\
 &+ \frac{W_c^{k_1/2N}}{W_s^{k_1/2N}} C(k'_1, k_2, \mathbf{n}) - \frac{W_c^{k_1/2N} W_s^{k_2/2N}}{W_s^{k_1/2N} W_c^{k_2/2N}} C(\mathbf{k}', \mathbf{n}),
 \end{aligned} \tag{3.46}$$

and

$$\begin{aligned}
 \frac{4}{W_s^{k_1/2N} W_s^{k_2/2N}} C(3, \mathbf{k}, \mathbf{n}) &= \frac{1}{\alpha_{k_1} \alpha_{k_2}} C(\mathbf{k}, \mathbf{n}) + \frac{W_c^{k_2/2N}}{W_s^{k_2/2N}} C(k_1, k'_2, \mathbf{n}) \\
 &+ \frac{W_c^{k_1/2N}}{W_s^{k_1/2N}} C(k'_1, k_2, \mathbf{n}) + \frac{W_c^{k_1/2N} W_c^{k_2/2N}}{W_s^{k_1/2N} W_s^{k_2/2N}} C(\mathbf{k}', \mathbf{n}),
 \end{aligned} \tag{3.47}$$

where the $C(\mathbf{k}, \mathbf{n})$ term on the RHS of (3.45) to (3.47) is the 2-D DCT-II, and the rest of the terms on the RHS of (3.45) to (3.47) are the aliased components. The complete derivation of the aliased components for the four subband transform is shown in Appendix A. The diminishing factor of $C(\mathbf{k}, \mathbf{n})$ in (3.45) to (3.47) is found to be different from (3.44), and it is clear that the diminishing factor of $C(\mathbf{k}, \mathbf{n})$ is the square of the twiddle factors of the subband transform. Using (3.45) to (3.47), the aliased components of $\mathbf{C}_n^{\text{ApD2}}$ to $\mathbf{C}_n^{\text{ApD7}}$ are obtained using the same approach to obtain the aliased component of $\mathbf{C}_n^{\text{ApD1}}$.

Table 3.5 Normalized aliased component at $C(\mathbf{k}', \mathbf{n})$ of $\mathbf{C}_n^{\text{ApD1}}$.

k'_1, k'_2	Aliasing at k'_1, k'_2			k'_1, k'_2	Aliasing at k'_1, k'_2
	$N=2$	$N=4$	$N=8$		
0, 0	0.000	0.000	0.000	4, 0	0.000
0, 1	0.000	0.000	0.000	4, 1	0.199
0, 2		0.000	0.000	4, 2	0.414
0, 3		0.000	0.000	4, 3	0.668
0, 4			0.000	4, 4	1.000
0, 5			0.000	4, 5	1.497
0, 6			0.000	4, 6	2.414
0, 7			0.000	4, 7	5.027
1, 0	0.000	0.000	0.000	5, 0	0.000
1, 1	1.000	0.172	0.039	5, 1	0.298
1, 2		0.414	0.082	5, 2	0.619
1, 3		1.000	0.133	5, 3	1.000
1, 4			0.199	5, 4	1.497
1, 5			0.298	5, 5	2.239
1, 6			0.480	5, 6	3.613
1, 7			1.000	5, 7	7.524
2, 0		0.000	0.000	6, 0	0.000
2, 1		0.414	0.082	6, 1	0.480
2, 2		1.000	0.172	6, 2	1.000
2, 3		2.414	0.277	6, 3	1.613
2, 4			0.414	6, 4	2.414
2, 5			0.619	6, 5	3.613
2, 6			1.000	6, 6	5.828
2, 7			2.082	6, 7	12.137
3, 0		0.000	0.000	7, 0	0.000
3, 1		1.000	0.133	7, 1	1.0000
3, 2		2.414	0.277	7, 2	2.082
3, 3		5.828	0.447	7, 3	3.359
3, 4			0.668	7, 4	5.027
3, 5			1.000	7, 5	7.524
3, 6			1.613	7, 6	12.137
3, 7			3.359	7, 7	25.274

A summary of the linear distortion and aliased components of the seven approximation of DCT-II is presented in Table 3.6. Since the six other approximations of DCT-II use different combinations of subband transforms, some of the aliased components might be partially or fully canceled due to the interaction of the aliased components of the four subband transforms. On this note, ApD2 and ApD3 have only one aliased component at $C(k'_1, k_2, \mathbf{n})$ and $C(k_1, k'_2, \mathbf{n})$, respectively. Again, we assume $C(\mathbf{k}, \mathbf{n})$ is a flat spectrum, the alias contribution of $C(k'_1, k_2, \mathbf{n})$, $C(k_1, k'_2, \mathbf{n})$, and $C(\mathbf{k}', \mathbf{n})$ of $\mathbf{C}_n^{\text{ApD2}}$ to $\mathbf{C}_n^{\text{ApD7}}$ are tabulated in Table 3.7, 3.8, and 3.9, respectively. From these tables, it is observed that the aliased components of $\mathbf{C}_n^{\text{ApD5}}$ to $\mathbf{C}_n^{\text{ApD7}}$ are generally smaller than those of $\mathbf{C}_n^{\text{ApD1}}$ to $\mathbf{C}_n^{\text{ApD4}}$.

To further analyze the performance of $\mathbf{C}_n^{\text{ApD1}}$ to $\mathbf{C}_n^{\text{ApD7}}$, the total alias contributed by the aliased component $C(k_1, k'_2, \mathbf{n})$, $C(k'_1, k_2, \mathbf{n})$, and $C(\mathbf{k}', \mathbf{n})$ is analyzed in the following. Since the MSE between $\mathbf{C}_n^{\text{ApD1}}$ to $\mathbf{C}_n^{\text{ApD7}}$ and $\mathbf{C}_n$ is dependent on the aliased component $C(k'_1, k_2, \mathbf{n})$, $C(k_1, k'_2, \mathbf{n})$, and $C(\mathbf{k}', \mathbf{n})$, which are contributed from the combination of subband transforms in $\mathbf{C}_n^{\text{ApD1}}$ to $\mathbf{C}_n^{\text{ApD7}}$, a lower total alias from these aliased components would suggest a better approximation of $\mathbf{C}_n$. From (3.43), the mean square alias (MSA) of ApD1 MSA_1 is computed as

$$MSA_1 = \frac{1}{N^2} \sum_{k_1=0}^{N-1} \sum_{k_2=0}^{N-1} \left(-\frac{W_c^{k_1/2N} W_s^{k_2/2N}}{W_c^{k_2/2N}} C(k_1, k'_2, \mathbf{n}) - \frac{W_s^{k_1/2N}}{W_c^{k_1/2N}} C(k'_1, k_2, \mathbf{n}) + \frac{W_s^{k_1/2N} W_s^{k_2/2N}}{W_c^{k_1/2N} W_c^{k_2/2N}} C(\mathbf{k}', \mathbf{n}) \right)^2, \quad (3.48)$$

Table 3.6 Alias contribution of approximation of DCT-II.

Approx. DCT-II	Linear Distortion/ $\alpha_{k_1} \alpha_{k_2}$	Aliased Component		
		$C(k_1, k'_2, \mathbf{n})/\alpha_{k_1} \alpha_{k_2}$	$C(k'_1, k_2, \mathbf{n})/\alpha_{k_1} \alpha_{k_2}$	$C(\mathbf{k}', \mathbf{n})/\alpha_{k_1} \alpha_{k_2}$
ApD1	$(W_c^{k_1/2N})^2 (W_c^{k_2/2N})^2$	$-(W_c^{k_1/2N})^2 W_c^{k_2/2N} W_s^{k_2/2N}$	$-W_c^{k_1/2N} W_s^{k_1/2N} (W_c^{k_2/2N})^2$	$W_c^{k_1/2N} W_c^{k_2/2N} W_s^{k_1/2N} W_s^{k_2/2N}$
ApD2	$(W_c^{k_1/2N})^2$	0	$-W_c^{k_1/2N} W_s^{k_1/2N}$	0
ApD3	$(W_c^{k_2/2N})^2$	$-W_c^{k_2/2N} W_s^{k_2/2N}$	0	0
ApD4	$(W_c^{k_1/2N})^2 (W_c^{k_2/2N})^2$ + $(W_s^{k_1/2N})^2 (W_s^{k_2/2N})^2$	$-W_c^{k_2/2N} W_s^{k_2/2N}$	$-W_c^{k_1/2N} W_s^{k_1/2N}$	$2W_c^{k_1/2N} W_c^{k_2/2N} W_s^{k_1/2N} W_s^{k_2/2N}$
ApD5	$(W_c^{k_1/2N})^2 + (W_s^{k_1/2N})^2 (W_c^{k_2/2N})^2$	$-(W_s^{k_1/2N})^2 W_c^{k_2/2N} W_s^{k_2/2N}$	$-W_c^{k_1/2N} W_s^{k_1/2N} (W_s^{k_2/2N})^2$	$-W_c^{k_1/2N} W_c^{k_2/2N} W_s^{k_1/2N} W_s^{k_2/2N}$
ApD6	$(W_c^{k_1/2N})^2 + (W_s^{k_1/2N})^2 (W_s^{k_2/2N})^2$	$(W_s^{k_1/2N})^2 W_c^{k_2/2N} W_s^{k_2/2N}$	$-W_c^{k_1/2N} W_s^{k_1/2N} (W_c^{k_2/2N})^2$	$W_c^{k_1/2N} W_c^{k_2/2N} W_s^{k_1/2N} W_s^{k_2/2N}$
ApD7	$(W_c^{k_2/2N})^2 + (W_s^{k_1/2N})^2 (W_s^{k_2/2N})^2$	$-(W_c^{k_1/2N})^2 W_c^{k_2/2N} W_s^{k_2/2N}$	$W_c^{k_1/2N} W_s^{k_1/2N} (W_s^{k_2/2N})^2$	$W_c^{k_1/2N} W_c^{k_2/2N} W_s^{k_1/2N} W_s^{k_2/2N}$

and MSA_2 to MSA_7 can be obtained using Table 3.6. The plots of MSA_2 to MSA_7 for a block size 2^N is presented in Fig. 3.7. The assumption of a flat DCT spectrum $C(\mathbf{k}, \mathbf{n})$ results in the following relation: $\mathbf{C}_n^{\text{ApD2}} = (\mathbf{C}_n^{\text{ApD3}})^t$ and $\mathbf{C}_n^{\text{ApD6}} = (\mathbf{C}_n^{\text{ApD7}})^t$. Therefore, the same amount of total alias is found in MSA_2 and MSA_3 , as well as MSA_6 and MSA_7 , as shown in Fig. 3.7. This same observation can be made from Table 3.7 to 3.9. Consistent with the observations made from Table 3.7 to 3.9, increasing N significantly increases the total alias of $\mathbf{C}_n^{\text{ApD1}}$ to $\mathbf{C}_n^{\text{ApD4}}$, while the total alias of $\mathbf{C}_n^{\text{ApD5}}$ to $\mathbf{C}_n^{\text{ApD7}}$ remains fairly constant for various values of N . For smaller block sizes, the total alias of $\mathbf{C}_n^{\text{ApD1}}$ to $\mathbf{C}_n^{\text{ApD7}}$ are quite similar, and $\mathbf{C}_n^{\text{ApD1}}$ is found to produce lesser alias as compared to $\mathbf{C}_n^{\text{ApD4}}$. Significant reduction of alias is obtained using $\mathbf{C}_n^{\text{ApD2}}$, $\mathbf{C}_n^{\text{ApD3}}$, and $\mathbf{C}_n^{\text{ApD5}}$ to $\mathbf{C}_n^{\text{ApD7}}$ as compared to $\mathbf{C}_n^{\text{ApD1}}$.

Fig. 3.7 Total alias of MSA_1 to MSA_7 for a block size 2^N .

Table 3.7 Normalized aliased component $C(k'_1, k_2, \mathbf{n})$ for $N = 4$.

k'_1, k_2	Aliased component $C(k'_1, k_2, \mathbf{n})$ for $k'_1, k_2 = 0, 1, \dots, N-1$				
	ApD2	ApD4	ApD5	ApD6	ApD7
$0, k_2$	0.000	0.000	0.000	0.000	0.000
$1, 0$	-0.414	-0.414	-0.053	-0.414	0.000
$1, 1$	-0.414	-0.333	-0.191	-0.345	0.059
$1, 2$	-0.414	0.000	-0.345	-0.191	0.308
$1, 3$	-0.414	1.000	0.000	-0.053	1.112
$2, 0$	-1.000	-1.000	-0.079	-1.000	0.000
$2, 1$	-1.000	-0.707	-0.333	-0.745	0.079
$2, 2$	-1.000	0.000	-0.745	-0.333	0.333
$2, 3$	-1.000	0.707	0.000	-0.079	0.745
$3, 0$	-2.412	-2.412	-0.059	-2.414	0.000
$3, 1$	-2.412	-1.000	-0.308	-1.112	0.053
$3, 2$	-2.412	0.000	-1.112	-0.380	0.191
$3, 3$	-2.412	0.333	0.000	-0.059	0.345

Table 3.8 Normalized aliased component $C(k_1, k'_2, \mathbf{n})$ for $N = 4$.

k_1, k'_2	Aliased component $C(k_1, k'_2, \mathbf{n})$ for $k_1, k'_2 = 0, 1, \dots, N-1$				
	ApD3	ApD4	ApD5	ApD6	ApD7
$k_1, 0$	0.000	0.000	0.000	0.000	0.000
$0, 1$	-0.414	-0.414	0.000	0.000	-0.414
$1, 1$	-0.414	-0.333	-0.053	0.059	-0.345
$2, 1$	-0.414	0.000	-0.191	0.308	-0.191
$3, 1$	-0.414	1.000	-0.345	1.112	-0.053
$0, 2$	-1.000	-1.000	0.000	0.000	-1.000
$1, 2$	-1.000	-0.707	-0.079	0.079	-0.745
$2, 2$	-1.000	0.000	-0.333	0.333	-0.333
$3, 2$	-1.000	0.707	-0.745	0.745	-0.079
$0, 3$	-2.414	-2.412	0.000	0.000	-2.414
$1, 3$	-2.414	-1.000	-0.059	0.053	-1.112
$2, 3$	-2.414	0.000	-0.308	0.191	-0.308
$3, 3$	-2.414	0.333	-1.112	0.345	-0.059

Table 3.9 Normalized aliased component $C(\mathbf{k}', \mathbf{n})$ for $N = 4$.

k'_1, k'_2	Aliased component $C(\mathbf{k}', \mathbf{n})$ for $k'_1, k'_2 = 0, 1, \dots, N - 1$			
	ApD4	ApD5	ApD6	ApD7
$0, k'_2$	0.000	0.000	0.000	0.000
$1, 0$	0.333	0.000	0.000	0.000
$1, 1$	0.707	-0.128	0.143	0.143
$1, 2$	1.000	-0.191	0.191	0.308
$1, 3$	0.000	-0.143	0.128	0.461
$2, 0$	0.707	0.000	0.000	0.000
$2, 1$	1.000	-0.191	0.308	0.191
$2, 2$	0.707	-0.333	0.333	0.333
$2, 3$	0.000	-0.308	0.191	0.308
$3, 0$	1.000	0.000	0.000	0.000
$3, 1$	0.707	-0.143	0.461	0.128
$3, 2$	0.333	-0.308	0.308	0.191
$3, 3$	0.000	-0.461	0.143	0.143

3.4 Simulation Results and Discussion

Until now, the aliased analysis of $\mathbf{C}_n^{\text{ApD1}}$ to $\mathbf{C}_n^{\text{ApD7}}$ are performed using a flat DCT spectrum. However, real world signals rarely possess a flat DCT spectrum, especially in the case of images. An $N \times N$ sub-image can be well approximated by various combinations of $c(0, \mathbf{q}, \mathbf{n})$, $c(1, \mathbf{q}, \mathbf{n})$, $c(2, \mathbf{q}, \mathbf{n})$, and $c(3, \mathbf{q}, \mathbf{n})$, depending on the content of the sub-image, which can be classified as a plain, edge or texture block as shown in Section 2.4. To approximate $\mathbf{C}_n$, the appropriate selection of $\mathbf{C}_n^{\text{ApD1}}$ to $\mathbf{C}_n^{\text{ApD7}}$ would be critical to keep aliasing at minimum while maintaining low computational cost.

An approximation of the DCT-II can be obtained by simply computing the subbands transforms of $c(0, \mathbf{q}, \mathbf{n})$, $c(1, \mathbf{q}, \mathbf{n})$, $c(2, \mathbf{q}, \mathbf{n})$, and $c(3, \mathbf{q}, \mathbf{n})$, which contain significant energy [MSP90, HH93]. This algorithm is referred as adaptive SBDCT (AdS1). In order to evaluate the performance of AdS1, we adopt the peak

signal-to-noise ratio (PSNR) and the structural similarity (SSIM) index [WBS04].

Let $c(\mathbf{x})$ and $\hat{c}(\mathbf{x})$ denote an image and its reconstructed image at $\mathbf{x}$, respectively,

where $\mathbf{x} = \{x_1, x_2\}$, $x_1 = 0, 1, \dots, H - 1$ and $x_2 = 0, 1, \dots, W - 1$. The PSNR

between the original image and its reconstructed image is given as

$$\text{PSNR} = 20 \log_{10} \frac{255}{\sqrt{\frac{1}{HW} \sum_{x_1=0}^{W-1} \sum_{x_2=0}^{H-1} |c(\mathbf{x}) - \hat{c}(\mathbf{x})|^2}}. \quad (3.49)$$

It is well known that PSNR correlates poorly with perceived image quality [Gir93, EF95], but this measure is still extensively used in image processing. This is largely attributed to its simple computational scheme. Recently, Wang *et al.* [WBS04] proposed the SSIM index, which compares the local patterns of pixel intensities after normalized for luminance and contrast. This index is developed under the assumption that the HVS is highly adapted to extract structural information from image and video contents. From their results, it is shown that the SSIM index correlates well with perceived quality measurement.

The luminance and contrast of an image are estimated using the mean intensity μ_c and standard deviation α_c , respectively. The SSIM map at $\mathbf{x}$ is computed as

$$\text{SSIM}(\mathbf{x}) = \frac{(2\mu_c(\mathbf{x})\mu_{\hat{c}}(\mathbf{x}) + a_1)(\alpha_{c\hat{c}}(\mathbf{x}) + a_2)}{(\mu_c^2(\mathbf{x}) + \mu_{\hat{c}}^2(\mathbf{x}) + a_1)(\alpha_c^2(\mathbf{x}) + \alpha_{\hat{c}}^2(\mathbf{x}) + a_2)}, \quad (3.50)$$

where $a_1 = (0.01M)^2$, $a_2 = (0.03M)^2$, and $M = 255$ for 8 bit grayscale images.

To avoid blocking artifacts in the SSIM map, Wang *et al.* computed the estimates of the local statistics as

$$\mu_{\hat{c}}(\mathbf{x}) = \sum_{m_1=0}^{N_{\text{ssim}}-1} \sum_{m_2=0}^{N_{\text{ssim}}-1} w_g(\mathbf{m}) c(x_1 + m_1, x_2 + m_2), \quad (3.51)$$

$$\sigma_c(\mathbf{x}) = \left[\sum_{m_1=0}^{N_{\text{ssim}}-1} \sum_{m_2=0}^{N_{\text{ssim}}-1} \left(c(x_1 + m_1, x_2 + m_2) - \mu_c(\mathbf{x}) \right)^2 \text{wg}(\mathbf{m}) \right]^{0.5}, \quad (3.52)$$

$$\begin{aligned} \sigma_{c\hat{c}}(\mathbf{x}) = & \sum_{m_1=0}^{N_{\text{ssim}}-1} \sum_{m_2=0}^{N_{\text{ssim}}-1} \left(c(x_1 + m_1, x_2 + m_2) - \mu_c(\mathbf{x}) \right) \\ & \cdot \left(\hat{c}(x_1 + m_1, x_2 + m_2) - \mu_{\hat{c}}(\mathbf{x}) \right) \text{wg}(\mathbf{m}), \end{aligned} \quad (3.53)$$

where $\text{wg}(\mathbf{m})$ is an $N_{\text{ssim}} \times N_{\text{ssim}}$ circular-symmetric Gaussian weighting function with standard deviation of 1.5 samples. In order to obtain a single distortion index of an image, the mean SSIM (MSSIM) can be used and is given as

$$\text{MSSIM} = \frac{1}{WH} \sum_{x_1=0}^{W-1} \sum_{x_2=0}^{H-1} \text{SSIM}(\mathbf{x}). \quad (3.54)$$

In the following, the performance of $\mathbf{C}_n^{\text{ApD1}}$ to $\mathbf{C}_n^{\text{ApD7}}$ is evaluated using 45 test images (TI-1 to TI-45) in Appendix B. The PSNR and MSSIM values between the original test-images and their corresponding reconstructed images are summarized in Table 3.10, and a scatter plot of the PSNR versus MSSIM values is shown in Fig. 3.8. For every $N \times N$ sub-image of an image, only the subband transform having the highest energy is computed in SBDCT. The percentage of $\mathbf{c}_{0,n}$ used in the $N \times N$ sub-images of 45 test-images is also presented in Table 3.10.

From Table 3.10, it is observed that at least 99.31% of $\mathbf{c}_{0,n}$ in the 8×8 sub-images of the test images contain the highest energy. This finding validates the assumption made by Jung *et al.* [JMM96] in the derivation of their FSA. However, less than 50% of the constructed images have PSNR values larger than 30 dB. In addition, we found that only 30% of the test-images have MSSIM values larger than 0.9, which translate to mean opinion scores (MOS) larger than 45 out of 80

Table 3.10 PSNR and MSSIM values of images reconstructed using AdS1.

Test Image	PSNR (dB)	MSSIM	$c_{0,n}$ (%)	Test Image	PSNR (dB)	MSSIM	$c_{0,n}$ (%)
TI-1	20.84	0.805	98.24	TI-24	29.61	0.915	99.92
TI-2	25.78	0.839	95.51	TI-25	27.65	0.903	99.92
TI-3	27.60	0.827	99.68	TI-26	34.82	0.951	100.00
TI-4	28.59	0.840	99.54	TI-27	34.78	0.929	100.00
TI-5	27.87	0.895	99.49	TI-28	30.62	0.916	99.79
TI-6	31.56	0.901	99.73	TI-29	35.86	0.958	100.00
TI-7	30.48	0.932	99.73	TI-30	32.89	0.931	100.00
TI-8	34.31	0.882	99.90	TI-31	29.15	0.903	99.92
TI-9	30.94	0.877	99.17	TI-32	29.85	0.917	99.87
TI-10	23.89	0.809	99.78	TI-33	25.15	0.774	96.31
TI-11	25.89	0.801	99.61	TI-34	32.36	0.871	100.00
TI-12	27.82	0.831	99.95	TI-35	32.78	0.917	98.91
TI-13	22.69	0.725	98.39	TI-36	23.13	0.731	98.58
TI-14	28.96	0.846	99.68	TI-37	24.75	0.824	99.97
TI-15	25.00	0.772	99.39	TI-38	26.65	0.816	97.80
TI-16	27.85	0.859	99.39	TI-39	30.70	0.922	99.45
TI-17	25.44	0.812	99.12	TI-40	22.60	0.782	99.05
TI-18	23.46	0.761	96.34	TI-41	29.53	0.850	99.74
TI-19	31.34	0.784	98.63	TI-42	27.35	0.864	99.30
TI-20	29.26	0.935	99.95	TI-43	28.13	0.837	99.35
TI-21	38.51	0.966	100.00	TI-44	31.14	0.889	99.95
TI-22	36.03	0.960	100.00	TI-45	27.66	0.828	99.84
TI-23	33.47	0.951	100.00	Mean	28.99	0.863	99.31

Fig. 3.8 Scatter plot of PSNR versus MSSIM of the images reconstructed using AdS1. The PSNR and MSSIM values are presented in Table 3.10.

[WBS04]. These observations, together with an average PSNR value of 28.99 dB and an average MSSIM value of 0.863, suggest that AdS1 leads to poor image quality. Comparing the PSNR values with MSSIM values for the 45 test-images, it is evident that PSNR are inconsistent with MSSIM, in particularly, TI-1, TI-19, TI-20, and TI-25 (red points in scatter plot shown in Fig. 3.8). For example, the reconstructed test-image TI-25 exhibits low PSNR values (< 30 dB) but high MSSIM index (> 0.9).

Figure 3.9 shows the original image, reconstructed image, error map, and the SSIM map of TI-25. The bright and dark regions of the SSIM map indicates low and high amount of distortion, respectively. The error map of TI-25 is the normalized difference between the original and reconstructed images. The normalized difference is then inverted so that '1' and '0' of the error map indicate that the pixel is identical and maximally different, respectively. From the reconstructed image, error map, and SSIM map, it is observed that the edges of the field and vegetation at the top part of TI-25, and the plants at the bottom of TI-25 are poorly reconstructed by AdS1. As the structure of these objects in the reconstructed image is preserved, the MSSIM value of the reconstructed image is relatively high (MSSIM = 0.903). However, this relatively high MSSIM value is not consistent with the low PSNR, which is based on the MSE across the whole reconstructed image. Even possessing a relatively high MSSIM value, Fig. 3.8(b) clearly displays the limitation of AdS1 in reconstructing edges found in TI-25.

As discussed in Section 3.3, the low average values of PSNR and MSSIM of the reconstructed images using AdS1 are attributed to the high aliased components found in AdS1. To mitigate this problem, more than one subband to approximate each 8×8 sub-image should be used to improve the performance of

Fig. 3.9 Reconstructing TI-25 using AdS1, (a) original image, (b) reconstructed image, (c) SSIM map, and (d) error map. These cropped images are un-scaled.

AdS1 and this algorithm is referred as AdS2. Using the subband having the highest energy as a reference, additional subbands are selected to improve the quality of the reconstructed images. While minimizing the number of subbands used, we empirically determined that additional subbands (up to three subbands including reference subband) should be selected if they contain more than 4% of the energy content of the subband having the highest energy.

The PSNR and MSSIM values of the images reconstructed using AdS2 are summarized in Table 3.11, and a scatter plot of the PSNR and MSSIM values is shown in Fig. 3.10. From Table 3.11, the means of PSNR and MSSIM of AdS2 are found to be higher than AdS1 by 4.35 dB and 0.062, respectively. This increase in average MSSIM value translates to an increase of MOS from 38 to 52, which suggests noticeable improvement in the images reconstructed by AdS2. The standard deviations of the PSNR and MSSIM of the images reconstructed using AdS1 are found to be reduced from 4.022 dB and 0.065 to 2.945 dB and 0.033, respectively, using AdS2. As compared to AdS1, AdS2 only requires a modest increase of 0.306 multiplications and 0.967 additions per pixel on the average.

Even though the standard deviations of the PSNR and MSSIM of the images reconstructed using AdS2 are reduced, three (TI-1, TI-19, and TI-25) out of the four images highlighted in Fig. 3.9 still remain at the outer region of the scatter plot of the images reconstructed using AdS2. A comparison of TI-1 and TI-25 reconstructed using AdS1 and AdS2 is shown in Fig. 3.11. As shown in Fig. 3.11(a), the reconstructed images of TI-1 and TI-25 using AdS2 are clearly superior than the ones from AdS1. For TI-1, a 7.38 dB improvement in PSNR is achieved using AdS2 as compared to AdS1. Yet, the PSNR of the reconstructed image from AdS2 remains low (28.22 dB) due to the numerous edges in TI-1, which led to aliasing in the reconstructed images from AdS1 and AdS2. Due to this reason, the PSNR of reconstructed image of TI-25 using AdS2 also remains at moderate level (32.48 dB) even though the MSSIM is relatively high (0.959).

Upon visual inspection, TI-19 is found to have a grainy texture throughout the whole image. Since MSSIM takes into account of differences of the standard

Table 3.11 PSNR and MSSIM values of images reconstructed using AdS2.

Test Image	PSNR (dB)	MSSIM	Subband Used in AdS2 (%)				Avg. Computation Cost (c_m, c_a)
			$c_{0,n}$	$c_{1,n}$	$c_{2,n}$	$c_{3,n}$	
TI-1	28.22	0.936	99.78	27.00	26.19	7.67	(89.28, 201.56)
TI-2	32.54	0.943	99.85	33.89	13.53	11.21	(88.36, 199.56)
TI-3	30.39	0.877	99.98	23.88	18.29	4.32	(83.51, 185.09)
TI-4	33.42	0.909	100.00	36.06	22.44	4.08	(90.39, 204.39)
TI-5	34.32	0.953	100.00	25.22	26.88	0.15	(85.94, 191.65)
TI-6	34.41	0.928	100.00	21.26	12.59	2.83	(79.43, 172.36)
TI-7	35.46	0.952	100.00	11.55	10.03	0.17	(73.14, 152.19)
TI-8	35.72	0.905	100.00	18.58	8.79	5.52	(77.87, 167.87)
TI-9	34.76	0.929	100.00	15.79	24.95	0.88	(81.48, 177.05)
TI-10	29.39	0.897	100.00	36.47	37.45	1.42	(95.64, 223.24)
TI-11	28.99	0.877	100.00	27.83	25.31	4.35	(88.19, 199.54)
TI-12	31.14	0.863	100.00	15.26	17.19	0.56	(77.88, 167.29)
TI-13	27.27	0.854	99.78	26.68	33.20	11.79	(93.81, 216.87)
TI-14	32.79	0.902	100.00	22.71	26.71	2.08	(85.68, 190.13)
TI-15	29.55	0.869	99.98	31.93	27.00	4.91	(90.77, 207.57)
TI-16	32.49	0.919	99.98	20.68	27.12	4.13	(85.79, 191.44)
TI-17	30.87	0.911	100.00	40.04	20.68	2.71	(90.65, 205.26)
TI-18	31.22	0.940	99.98	46.26	61.67	11.08	(113.97, 279.89)
TI-19	33.07	0.852	99.78	11.65	26.03	23.46	(89.85, 204.50)
TI-20	34.99	0.960	100.00	13.13	14.04	0.00	(75.41, 159.32)
TI-21	39.79	0.969	100.00	7.214	6.51	1.25	(70.30, 144.86)
TI-22	38.20	0.966	100.00	6.77	4.38	0.10	(68.73, 139.73)
TI-23	36.47	0.961	100.00	7.318	8.65	0.03	(70.72, 145.60)
TI-24	33.35	0.939	100.00	10.21	14.40	0.00	(74.34, 156.65)
TI-25	32.48	0.959	100.00	26.48	38.20	1.43	(91.79, 210.04)
TI-26	37.53	0.964	100.00	9.896	11.64	0.03	(73.06, 152.46)
TI-27	37.99	0.955	100.00	16.30	18.09	0.08	(78.48, 169.86)
TI-28	35.71	0.958	100.00	20.83	30.47	0.08	(85.58, 189.68)
TI-29	38.79	0.968	100.00	8.23	6.589	0.49	(70.43, 144.84)
TI-30	34.80	0.939	100.00	4.56	6.33	0.00	(68.57, 139.08)
TI-31	32.46	0.951	100.00	24.71	40.31	0.00	(91.31, 207.38)
TI-32	35.13	0.954	100.00	12.92	19.43	0.00	(77.58, 164.27)
TI-33	33.09	0.938	100.00	48.67	57.15	5.81	(110.89, 270.05)
TI-34	34.39	0.889	100.00	3.47	4.56	0.24	(67.47, 136.03)
TI-35	37.21	0.959	100.00	13.13	21.21	4.89	(80.52, 176.13)
TI-36	28.89	0.873	99.98	35.73	50.99	8.79	(104.11, 249.89)
TI-37	28.93	0.896	100.00	23.24	27.26	1.97	(86.08, 191.76)
TI-38	33.41	0.934	100.00	17.04	45.57	6.02	(92.84, 212.36)
TI-39	34.01	0.953	99.97	20.36	24.20	5.19	(84.89, 188.85)
TI-40	30.57	0.924	100.00	41.75	38.54	2.82	(98.91, 229.83)
TI-41	33.81	0.908	100.00	21.63	18.77	1.99	(81.83, 179.87)
TI-42	32.48	0.929	99.93	23.78	35.40	8.39	(92.34, 212.56)
TI-43	33.05	0.930	100.00	18.57	42.89	3.66	(91.37, 206.43)
TI-44	35.14	0.913	100.00	6.72	8.45	0.13	(70.43, 144.29)
TI-45	31.69	0.899	100.00	19.74	34.05	2.21	(87.52, 196.55)
Mean	33.34	0.925	99.98	22.23	24.31	3.53	(84.60, 187.91)

Fig. 3.10 Scatter plot of PSNR versus MSSIM of the images reconstructed using AdS2. The PSNR and MSSIM values are presented in Table 3.11.

Fig. 3.11 Cropped images of original image (left), reconstructed image using AdS1 (middle), reconstructed image using AdS2 (right) for (a) TI-1 and (b) TI-25. These cropped images are un-scaled.

deviation of the original and reconstructed image, MSSIM is relatively sensitive to the changes of the grainy texture found in TI-19. This sensitivity prevents high MSSIM to be obtained from the image reconstructed by AdS1 and AdS2. Among the test-images, TI-8 is also found to possess similar grainy texture in the background of the image. Figure 3.12 compares the SSIM maps of TI-8 and TI-9 reconstructed using AdS1 and AdS2. From the SSIM maps of the reconstructed images, it is clear that the differences in the grainy texture between the test-images and their reconstructed images led to low SSIM values in the SSIM maps, and eventually low MSSIM values.

A few test-images did not exhibit significant increase in PSNR and MSSIM when reconstructed using AdS2 as compared to AdS1. One example is TI-21, and the SSIM and error maps of the reconstructed TI-21 using AdS1 and AdS2 are presented in Fig. 3.13. As concluded in Section 3.3, AdS1 performs well if it is applied onto an image that do not contain a lot of edges, which is the case for TI-21. This characteristic of TI-21 led to minimum differences in terms of PSNR and MSSIM between the images reconstructed using AdS1 and AdS2, as shown in Fig. 3.13.

3.5 Conclusions

This chapter reviewed the derivation of the SBDCT, and subsequently its fast algorithm. Previously, the SBDCT is mainly derived using the row-column decomposition methods, which lead to inefficient computational structures [JMM96, TGW05b]. A new FSA based on the $N/2 \times N/2$ fast DCT-II has been presented in this chapter. Using the 2-D symmetry in the SBDCT coefficients as well as the zero SBDCT coefficients, we have shown that it is possible to compute

(a)

(b)

(c)

(d)

(e)

(f)

Fig. 3.12 Test images of (a) TI-8 and (b) TI-19, and SSIM maps of reconstructed images of (c) TI-8 using AdS1, (d) TI-19 using AdS1, (e) TI-8 using AdS2, and (f) TI-19 using AdS2.

(a)

(b)

(c)

(d)

(e)

Fig. 3.13 Error and SSIM maps of reconstructed images of TI-21. (a) Original image of TI-8. (b) SSIM maps of reconstructed images of TI-21 using AdS1 and AdS2 are (b) and (c), respectively. Error maps of reconstructed images of TI-21 using AdS1 and AdS2 are (d) and (e), respectively.

the 8×8 SBDCT at a computational cost of (204, 536) as compared to (432, 544) in [JMM96].

Due to the subband decomposition of the $N \times N$ sub-images in SBDCT, it is found that SBDCT is well-suited for parallel implementation, and is implemented on a real-time digital signal processor (DSP) [TGW05b] as well as on a mobile device platform [YTG07]. This particular characteristic of SBDCT leads to seven approximation of the 2-D DCT-II algorithm, which are obtained using different combinations of the subband transforms of the SBDCT. The simplest approximation of 2-D DCT-II using SBDCT is obtained by only computing the subband transform of $\mathbf{c}_{0,n}$. However, if there is significant energy in the aliased components, as shown in (3.42) and (3.44) for the 1-D and 2-D SBDCT, respectively, this approximation leads to perceptible distortion (aliasing in the edges) in the reconstructed image. For such cases, a good approximation of the DCT-II approximation would require more than one subband to avoid such distortion. An alternative approach AdS2, which compares the energy of $\mathbf{c}_{1,n}$, $\mathbf{c}_{2,n}$, and $\mathbf{c}_{3,n}$ with $\mathbf{c}_{0,n}$ is discussed. From the results in Section 3.4, this approach avoids the significant amount of distortion in the reconstructed image. Since a simple energy comparison may not correlate well with the human perception, the selection of 2-D DCT-II approximation should be performed by analyzing the energy of $\mathbf{c}_{0,n}$, $\mathbf{c}_{1,n}$, $\mathbf{c}_{2,n}$, and $\mathbf{c}_{3,n}$ with respect to the HVS, and shall be discussed in Chapter 4.

Chapter 4

Perceptual Image Coding with SBDCT and JND Modeling

Lossless compression typically offers compression ratio at the order of 3:1 [WSS00]. Since limited storage and transmission bandwidth are available, such compression ratio is inadequate for most applications. While lossy compression overcomes this problem, overly compressed JPEG and JPEG 2000 images exhibit various artifacts, such as blocking and blurring, respectively.

At the same level of compression, JPEG 2000 [CSE00] images do not exhibit the same level of degradation as compared to JPEG [Wal92] images. This is attributed to the discrete wavelet transform (DWT) and DCT adopted by JPEG 2000 [ISO00] and JPEG [ISO94] standards, respectively. At high compression ratio, content of JPEG 2000 images can be easily recognized as compared to JPEG images. Furthermore, JPEG 2000 offers PSNR and resolution scalability, which can be useful in many applications. The main disadvantage of JPEG 2000 is its higher complexity as compared to other image compression standards [SE00]. To date, DCT is still popular and used in many image and video compression standards, such as JPEG, MPEG-1/2/4, and H.261/3. Therefore, this chapter focuses on proposing a new perceptually-tuned image coder that is based on a variant of DCT, specifically, the SBDCT.

The key idea of perceptually-tuned image compression is to improve coding efficiency of images by exploiting the limitations of HVS. The physiological and

psychological mechanisms of HVS prevents it from sensing all changes in an image. Specifically, any changes in an image that are lower than the JND threshold are undetectable by the HVS. Over years, several computational models for JND have been proposed and these models are computed from subbands [SJ89, Wat93, TS96, HK00, HK02, ZLX05, ZLX08, WN09] or pixels [CL95, CC96, CB99, YLL03, YLL05, LLP10] of an image.

The JPEG standard specifies the DCT-II in its encoders and decoders. However, in our PIC, we have adopted the SBDCT [JMM96], which is derived from the DCT-II. Even though SBDCT is mathematically equivalent to DCT-II, SBDCT and DCT-II possess quite different computational structures. SBDCT operates on subbands of an $N \times N$ input sequence, whereas DCT-II operates directly on the $N \times N$ input sequence. To reduce the computational cost of DCT-II, Jung *et al.* [JMM96] approximated the DCT-II using ApD1, as discussed in Section 3.2.2. It is found that this approach reduces the computational cost of their image coder and the bit-rate of the compressed image, while reasonable image quality (PSNR < 32 dB) is maintained.

This chapter derives and analyzes a PIC that applies computational models of JND that are based on subbands and pixels of an image. A computational model of JND is usually used to derive a set of JND threshold in the pixel or subband domain, which determines the amount of quantization noise that can be added without degrading the quality of the compressed images. In the proposed PIC, a JND model based on pixels is adapted to take into account the computational structure of SBDCT, and applies the block classification results that are derived from SBDCT. Zhang *et al.* [ZLX08] highlighted the limitations of a pixel-based JND model in modeling the frequency and orientation selectivity of the HVS. This

limitation is avoided in the proposed PIC, which includes a DCT subband-based JND model in the quantization stage of the PIC. From our subjective experiments, it is verified that this unique integration of pixel-based and DCT subband-based JND models in a PIC leads to good compression performance while avoids high computational cost.

The rest of this chapter is organized as follows. First, the proposed PIC is introduced in Section 4.1. This is followed by a discussion on the various modules of the proposed PIC. This section also deals with the integration of the SBDCT and the JND models, which leads to the proposed PIC that takes advantage of the computational structure of SBDCT. Section 4.2 is concerned with the validation of the proposed PIC, which includes subjective evaluation of the proposed JND model. Next, a performance comparison of various PICs using subjective evaluation is discussed in Section 4.3. Finally, our conclusions are presented in Section 4.4.

4.1 Proposed Encoder

Chitprasert and Rao [CR90] computed a set of JND threshold that is used to quantize $N \times N$ DCT coefficients with respect to the spatial frequency sensitivity of the HVS. Similarly, Watson [Wat93] used a CSF [AP92] to compute a set of JND thresholds for the $N \times N$ DCT-II. It is well known that the HVS adapts to the local activities of an image, hence, using the same JND threshold for the whole image is not a good approach to incorporate the HVS into any imagery system. On the other hand, a different quantization matrix for every $N \times N$ DCT coefficients would lead to significant amount of side information, and increases the bit-rate of the compressed image. Watson [Wat93] avoided this problem by designing a

single quantization matrix, which minimize the overall perceptible distortion. Höntsch and Karam [HK00, HK02] estimated the quantization matrix using the encoded and predicted DCT-II coefficients. To tackle this issue on local activities of an image, we applied a JND model to determine the perceptually significant subbands of each $N \times N$ sub-image. By only operating on the perceptually significant subbands using SBDCT, computational cost of SBDCT and perceivable distortion in the compressed image are minimized [TGW05a, TG11]. In Chapter 3, it is shown that the AdS2 is capable of producing visually good images. Hence, this approach of selecting the perceptually significant subbands of each $N \times N$ sub-image based on the HVS can be viewed as an extension of AdS2.

Figure 4.1 illustrates the proposed PIC, which is perceptually-tuned by pixel-based and DCT subband-based JND models. An input image is divided into $N \times N$ sub-images, and then each $N \times N$ sub-image is further decomposed into M^2 subbands. In order to determine the perceptually significant subbands of each sub-image, the JND profile of the input image is also decomposed into M^2 subband JND profiles. Subsequently, each subband JND profile is divided into $(N/M) \times (N/M)$ blocks. This is followed by a energy comparison between the subband of an $N \times N$ sub-image and its subband JND profile. Any subband having lower energy than its corresponding $(N/M) \times (N/M)$ block of the subband JND profile is classified as perceptually insignificant, and is removed from each $N \times N$ sub-image (i.e., these subbands will not be processed in the later stages of the PIC). Since image are generally non-stationary, the viewer would be able to tolerate different amount of distortion in each $N \times N$ sub-image, depending on its content.

Fig. 4.1 Block diagram of the proposed PIC.

In other words, we expect each $N \times N$ sub-image to contain different combinations of perceptually significant subbands.

With this in mind, the combination of perceptually significant subbands of each $N \times N$ sub-image can be used to select an approximation method listed in Table 3.3 to approximate DCT-II. By approximating DCT-II based on human perception, this leads to lesser distortion perceived by a human receiver, as compared to AdS1 and AdS2. As the selection of subband is based on human perception, this approximation of DCT-II is referred as perceptual SBDCT (PSBDCT).

A spatial JND profile of an image usually comprises of two components [CL95, YLL05], namely, luminance adaptation and contrast masking. To estimate the spatial JND profile, a non-linear additivity model for masking (NAMM) [YLL05] is used to determine the combined effect of luminance adaptation and contrast masking. To estimate contrast masking, Yang *et al.* [YLL05] stated that there is a need to differentiate between edge and texture, and provided the following reasons. An edge is structurally simpler than a texture, and edges tend to be easily recognized by the HVS or a typical observer would have prior knowledge of how an edge looks like [EB98]. Moreover, HVS has acute sensitivity at or near the luminance edge [Gir93]. Based on Eckert's observations in [EB98], the JND threshold at a texture region is three times higher than those in an edge region. Therefore, classification of plain, edges and texture blocks should be performed to effectively estimate contrast masking in any image [YLL05, ZLX05, ZLX08, WN09, LLP10]. The proposed PIC performs block classification using an edge map, which is estimated from the high frequency components of each $N \times N$ sub-image.

The JND profile of an input image is decomposed into M^2 and N^2 subband JND to determine the perceptually significant subbands of each $N \times N$ sub-image, and to compute the quantization matrix for an $N \times N$ DCT block, respectively. To prevent over-estimating the spatial JND profile of an image, where perceptually insignificant subbands are removed from its $N \times N$ sub-images, the spatial JND profile at these locations are reduced accordingly. This updated spatial JND model together with a MTF is then used to derive a single quantization matrix. Similar to Watson's DCTune, this single quantization matrix is designed to minimize the overall perceptual distortion in an image. The final stages of the proposed PIC involves transformation of the perceptually significant subbands into the DCT domain, quantization, and entropy coding.

4.1.1 Spatial Just Noticeable Difference Model

In general, luminance adaptation and contrast masking are considered in most computational models for spatial JND. Let $t_{\text{NAMM}}(\mathbf{x})$ denote the spatial JND threshold of a pixel located at $\mathbf{x}$, where $\mathbf{x} = \{x_1, x_2\}$, $x_1 = 0, 1, \dots, H-1$, and $x_2 = 0, 1, \dots, W-1$. For achromatic images, a generalized expression for spatial JND using the NAMM [YLL03, YLL05] is given as

$$t_{\text{NAMM}}(\mathbf{x}) = t_{la}(\mathbf{x}) + t_{cm}(\mathbf{x}) - C_{ol} \times \min\{t_{la}(\mathbf{x}), t_{cm}(\mathbf{x})\}, \quad (4.1)$$

where $t_{la}(\mathbf{x})$ and $t_{cm}(\mathbf{x})$ are the visibility thresholds for luminance adaptation and contrast masking, respectively; C_{ol} accounts for the reduction of spatial JND due to overlapping effect in masking, where $0 < C_{ol} \leq 1$. It can be shown that the pixel-based JND models in [CL95, CC96, CB99] are special cases of (4.1).

Chou and Li proposed a pixel-based JND model in [CL95, CC96] that assumes a simplified relationship between luminance adaptation and contrast masking. By letting $C_{oe} = 1$ in (4.1), their pixel-based JND model is expressed as

$$t_{\text{chou}}(\mathbf{x}) = \max\{t_{la}(\mathbf{x}), t_{cm}(\mathbf{x})\}. \quad (4.2)$$

Chiu and Berger [CB99] considered that the luminance adaptation is the major contributor in their pixel-based JND model, which is given as

$$t_{\text{chiu}}(\mathbf{x}) = t_{la}(\mathbf{x}) + Ct_{cm}(\mathbf{x}), \quad (4.3)$$

where C is experimentally found to be between 0.5 and 1. In comparison to the Chou's JND model [CL95], Yang *et al.* [YLL03, YLL05] and Liu *et al.* [LLP10] showed that NAMM leads to visually better images using subjective experiments.

For the proposed PIC, we adopt the NAMM to compute the JND profile of an image, and luminance adaptation is computed using (2.23). Contrast masking is determined as

$$\mathbf{t}_{cm} = \mathbf{w}_{cm} * \mathbf{h}, \quad (4.4)$$

where $*$ denotes convolution, $\mathbf{w}_{cm}$ is a set of weights relating to plain, edge and texture regions of an image, and $\mathbf{h}$ is a Gaussian low-pass filter to smoothen $\mathbf{w}_{cm}$. This low-pass filter has a kernel size of 7×7 and a standard deviation of 0.8 [YLL05].

4.1.2 Block Classification of $N \times N$ Sub-Image

Liu *et al.* [LLP10] used a total-variation based image decomposition [YGO05] to decompose an image into structural and texture components, which are used to estimate contrast masking contributed by edges and textures, respectively. This approach requires an estimate of the lowest correlation between the structural and

texture components, therefore, this approach can be computation intensive. Zhang *et al.* [ZLX05, ZLX08] classify each $N \times N$ DCT block into plain, edge and texture regions according to the energy computed from the DCT coefficients. While this approach is computational simpler as compared to the total-variation based decomposition, several thresholds in their block classification method have to be predetermined for different values of N . The block classification for $N = 8$ is reported in [ZLX05, ZLX08]. Wei and Ngan [WN09] performed block classification using the mean of edge pixels of an $N \times N$ block, therefore, their approach is invariant to the size of the DCT-II transform.

Wei and Ngan, Yang *et al.*, and Liu *et al.* used the Canny edge detector [Can86] to derive an edge map of an image, which is used to determine the edges and textures regions of an image. While the Canny edge operator is optimal in detecting edges, it is not designed to detect texture regions of an image, and it is computation intensive as compared to most edge operators. To avoid the high computational cost associated to Canny edge operator, we estimated an edge map of an image using the Hadamard transform found in SBDCT, which is defined as $\mathbf{P}_1$ of (3.10). Each $N \times N$ sub-image is decomposed into M^2 subbands using the Hadamard transform. In Chapter 2, we have considered $M = 2$, and leads to four subbands. The subbands $\mathbf{c}_{1,\mathbf{n}}$, $\mathbf{c}_{2,\mathbf{n}}$, and $\mathbf{c}_{3,\mathbf{n}}$ contain high frequency components of an $N \times N$ sub-image in the horizontal, vertical, and diagonal directions, respectively. Let $\mathbf{e}_{\text{map},\mathbf{n}}$ denote an $N \times N$ edge map of the $\mathbf{n}$ th sub-image. Using these subbands, an edge map of an image can be estimated by summing these subbands and is computed by

$$\mathbf{e}_{\text{map},\mathbf{n}} = (\mathbf{I}_4 \otimes \mathbf{P}_1)^t \mathbf{P}_2^t (\mathbf{c}_{1,\mathbf{n}} + \mathbf{c}_{2,\mathbf{n}} + \mathbf{c}_{3,\mathbf{n}}) \mathbf{P}_2 (\mathbf{I}_4 \otimes \mathbf{P}_1). \quad (4.5)$$

Let $e_{\text{map}}(\mathbf{p}, \mathbf{n})$ denote the elements of $\mathbf{e}_{\text{map}, \mathbf{n}}$. A binary edge map $e_{\text{b_map}}(\mathbf{p}, \mathbf{n})$ of the $\mathbf{n}$ th sub-image is computed as

$$e_{\text{b_map}}(\mathbf{p}, \mathbf{n}) = \begin{cases} 1, & e_{\text{map}}(\mathbf{p}, \mathbf{n}) > \alpha_{\text{edge}}, \\ 0, & \text{Otherwise,} \end{cases} \quad (4.6)$$

where $\alpha_{\text{edge}} = 10$ is found to be suitable for most of our test-images. Following the idea of Wei and Ngan, we perform the block classification of each $N \times N$ sub-image using the following:

$$bt(\mathbf{n}) = \begin{cases} \text{Plain,} & \text{for } \bar{P}_{\text{edge}} \leq \alpha_{\text{pixel}}, \\ \text{Edge,} & \text{for } 0.1 < \bar{P}_{\text{edge}} \leq \beta_{\text{pixel}}, \\ \text{Texture,} & \text{for } \bar{P}_{\text{edge}} > \beta_{\text{pixel}}, \end{cases} \quad (4.7)$$

where $\bar{P}_{\text{edge}}(\mathbf{n}) = e_{\text{b_map}}(\mathbf{p}, \mathbf{n}) / N^2$, α_{pixel} and β_{pixel} are empirically determined as 0.1 and 0.25, respectively.

The edge maps of "Barbara" and "Baboon" images obtained using (4.5), (4.6), and Canny edge operator are shown in Fig. 4.2. The block classification results of "Barbara" and "Baboon" images are shown in Fig. 4.3. It is clear that the Canny edge operator exhibits superior performance in terms of edge detection. However, this does not necessarily translate to superior block classification results since the Canny edge operator does not differentiate texture and edge regions. As compared to Zhang's block classification method, the block classification method proposed by Wei and Ngan is found to produce more consistent results in the bookshelf and table regions of the "Barbara" image. However, Wei's method is overly sensitive in the head, body, and pants regions of the "Barbara" image. On this note, Zhang's method is clearly superior than Wei's method in detecting the texture regions in head, body, and pants regions of the "Barbara" image. The same observations are also made for the results of proposed method, as shown in Fig. 4.3(e). However,

Fig. 4.2 Edge map of "Barbara" and "Baboon" images obtained using (a) to (b) subbands $c_{1,n}$, $c_{2,n}$, and $c_{3,n}$, (c) to (d) subbands $c_{1,n}$, $c_{2,n}$, and $c_{3,n}$ with thresholding, and (e) to (f) Canny edge operator with threshold set to 0.1.

Fig. 4.3 Block classification results of "Barbara" and "Baboon" images using (a) to (b) Wei's method, (c) to (d) Zhang's method, and (e) to (f) proposed method. Black, gray and white regions indicate plain, texture and edge regions, respectively.

the vertical and horizontal lines at the right side of the bookshelf are detected by Wei's and proposed methods are undetected by Zhang's method.

For the "Baboon" image, the majority of the image is classified as texture by Wei's and proposed methods, which is consistent with human evaluation. Wei's method also detected the fine edges at the bottom left and right sides of "Baboon" image, which are classified as plain regions by Zhang's and proposed methods. It is clear that the proposed method is relatively consistent with the texture detection for this image. Furthermore, the proposed method is most successful in detecting the small plain regions around the eyes. Hence, the proposed method possesses superior performance as compared to Wei's and Zhang's methods. Also, the proposed block classification does not require additional computational cost for computing subbands $\mathbf{c}_{1,\mathbf{n}}$, $\mathbf{c}_{2,\mathbf{n}}$, and $\mathbf{c}_{3,\mathbf{n}}$ since these subbands are part of the results of the SBDCT. Based on the results from the block classification method, the edge related weight $w_{cm}(\mathbf{n})$ is computed as

$$w_{cm}(\mathbf{n}) = \begin{cases} 0, & \text{for } bt(\mathbf{n}) = \text{Plain}, \\ \frac{\alpha_{cm}}{3}, & \text{for } bt(\mathbf{n}) = \text{Edge}, \\ \alpha_{cm}, & \text{for } bt(\mathbf{n}) = \text{Texture}, \end{cases} \quad (4.8)$$

where α_{cm} is 0.117 [YLL05].

4.1.3 Decomposition of JND Profile of an Image

As shown in Section 2.6, M^2 subband JND profiles can be estimated from a spatial JND profile [CL95, CC96]. The weighting factor $w_{M^2}(b_i)$ of (2.42) is derived from a MTF proposed by Mannos and Sakrison [MS74], and the weighting factor can be computed by [CL95]

$$w_{M^2}(b_i) = \left(d_{M^2}(b_i) \sum_{a_1=0}^{M^2-1} d_{M^2}^{-1}(a_1) \right)^{-1}, \quad (4.9)$$

where $b_i = 0, 1, \dots, M^2 - 1$. Sensitivity $d_{M^2}(b_i)$ is computed as

$$d_{M^2}(b_i) = \sum_{a_1=0}^{W/M-1} \sum_{a_2=0}^{H/M-1} MTF \left(\frac{W}{M} \bmod_M(b_i) + a_1, \frac{W}{M} \left\lfloor \frac{b_i}{M} \right\rfloor + a_2 \right), \quad (4.10)$$

where $\bmod_b(a)$ produces the remainder of a divided by b . Based on the assumption that the HVS is isotropic, a parametric form of a MTF at $\mathbf{x}$ is expressed as

$$MTF(\mathbf{x}) = a_{\text{MTF}} (b_{\text{MTF}} + c_{\text{MTF}} f(\mathbf{x})) \exp \left((-c_{\text{MTF}} f(\mathbf{x}))^{d_{\text{MTF}}} \right), \quad (4.11)$$

where $f(\mathbf{x})$ is the spatial frequency in cpd; a_{MTF} , b_{MTF} , c_{MTF} , and d_{MTF} are the parameters of the MTF. In [MS74], Mannos and Sakrison empirically obtained the following MTF:

$$MTF_{\text{MS}}(\mathbf{x}) = 2.6 (0.0192 + 0.114 f(\mathbf{x})) \exp \left\{ - (0.114 f(\mathbf{x}))^{1.1} \right\}. \quad (4.12)$$

The spatial frequency is expressed as $f(\mathbf{x}) = \sqrt{(x_1/v_{x_1})^2 + (x_2/v_{x_2})^2}$, where

$$\begin{aligned} v_{x_1} &= 2 \operatorname{atan}(\Lambda_{x_1}/2d), \\ v_{x_2} &= 2 \operatorname{atan}(\Lambda_{x_2}/2d), \end{aligned} \quad (4.13)$$

Λ_{x_1} , and Λ_{x_2} are the height and width of a pixel, respectively. Since the viewing distance d is defined as six times of the image height, we can compute the visual angle vertically as $v_{x_1} = 2 \operatorname{atan}(1/6144) = 0.0187^\circ$. If $\Lambda_{x_1} = \Lambda_{x_2}$, then we have $v_{x_1} = v_{x_2}$, and 53.76 pixels subtended 1° , vertically and horizontally, for a viewing distance at six times of the image height. For $M = 2$, and $W = H = 512$, the

weighting factor for the MTF $w_4(b_i)$ is found to be $\{0.1179, 0.2223, 0.2223, 0.4376\}$.

4.1.4 Computing Base Detection Threshold from MTF

As discussed by Nill [Nil85], MTFs that are obtained using sine or square grating functions might not yield good estimation of the HVS when applied to DCT-II as DCT-II implies an even extension of its input sequence. To avoid this issue, Nill proposed to scale such MTFs with

$$|A(f)| = \sqrt{\frac{1}{4} + \frac{1}{\pi^2} \left[\ln \left(\left(\frac{2\pi}{\alpha} f \right) + \sqrt{\frac{4\pi^2}{\alpha^2} f^2 + 1} \right) \right]^2}, \quad (4.14)$$

where

$$\alpha^{-1} = 2 \sin \left(\frac{1.5}{2\sqrt{0.5625 + d^2}} \right). \quad (4.15)$$

On the other hand, Chitprasert and Rao [CR90], as well as Wei and Ngan [WN09] had empirically obtained MTFs that are optimized for DCT-II.

Based on a set of experimental data from ModelFest [WA05] (uses mainly Gabor and Gaussian stimulus for subjective evaluation), it is found that the MTF peaks around 3 to 4 cpd. It is also noted that the stimulus presented to subjects in the ModelFest experiments subtended to 2.133° (falls within the foveal region), and the viewing of the stimulus is in binocular vision (both eyes are used to view stimulus). Despite the experiments in [WA05] are not targeted for DCT based image processing, it is still a good reference due to the extensive experimentation that led to the creation of ModelFest. Furthermore, Navarro *et al.* reported in [NAW93] that a MTF remains fairly constant for wide visual field up to 120° ,

therefore, it is reasonable to apply the observations from ModelFest into our discussion.

A comparison of the MTFs from Mannos and Sakrison [MS74], weighted MTF from Mannos and Sakrison, Nill [Nil85], Ngan *et al.* [NLS86], Chitprasert and Rao [CR90], as well as Wei and Ngan [WN09] is shown in Fig. 4.4. These MTFs are referred as MTF_{MS} , MTF_{WMS} , MTF_N , MTF_{NG} , MTF_{CR} , and MTF_{WN} , respectively. MTF_{WMS} is obtained by scaling MTF_{MS} using (4.14). The parameters of the various MTFs are summarized in Table 4.1. Since MTF_{CR} , and MTF_{WN} are measured using DCT basis vectors, there is no need to scale the MTF using (4.14).

Table 4.1 Parameters of various MTFs.

MTF	a_{MTF}	b_{MTF}	c_{MTF}	d_{MTF}
Mannos and Sakrison	2.600	0.019	0.114	1.100
Nill	2.500	0.080	0.180	1.000
Ngan <i>et al.</i>	2.380	0.130	0.290	1.000
Chitprasert and Rao	2.460	0.100	0.250	1.000
Wei and Ngan	0.610	2.176	0.180	1.000

The spatial frequency is related to the $\mathbf{k}$ th DCT subband by [CR90]

$$f_{DCT}(\mathbf{k}) = \frac{1}{2N} \sqrt{\left(k_1/v_{x_1}\right)^2 + \left(k_2/v_{x_2}\right)^2}, \quad (4.16)$$

and the peaks of MTF_{MS} , MTF_{WMS} , MTF_N , MTF_{NG} , MTF_{CR} , and MTF_{WN} are found to be 6.72, 10.08, 6.72, 3.36, 3.36, and 0 cpd, respectively. It is interesting to note that there is an increase of the response of MTF_{WN} as the spatial frequency is reduced from 3.36 to 0 cpd. This observation of MTF_{WN} does not agree well with the rest of the MTFs and the ModelFest data. However, due to the relatively flatness of MTF_{WN} within 0 to 3.36 cpd, this MTF is capable of compensating for

Fig. 4.4 Comparison of various MTFs for $N = 8$.

the change of base detection threshold when the viewer moves nearer to an image (causes the peak of the MTF shifts to the left).

Considering the MTF has a peak within 3 to 4 cpd, MTF_{NG} and MTF_{CR} should be used to determine the base detection threshold. Since the base detection threshold is inversely proportional to the MTF, therefore, MTF_{CR} is selected over MTF_{NG} to avoid high base detection threshold. As highlighted in [WA05], the accuracy of the MTF can be further enhanced by considering the oblique effect [BKW75, PW84, AP92] and the spatial summation effect [PAW93a].

The oblique effect of two DCT basis vectors is first discussed in [AP02], which involves scaling the first component on the RHS of (2.13) with the term $[r + (1-r)\cos^2 \theta(\mathbf{k})]$. Watson and Ahumada [WA05] also suggested to combine the oblique effect with the MTF in the spatial frequency domain. Hence, we compute the base detection threshold using the following equation [WN09]:

$$T_{b,\text{MTF}}(\mathbf{k}) = \frac{1}{\alpha_{k_1}} \frac{1}{\alpha_{k_2}} \frac{s \exp\left((c_{\text{MTF}} f(\mathbf{k}))^{d_{\text{MTF}}}\right)}{a_{\text{MTF}} (b_{\text{MTF}} + c_{\text{MTF}} f(\mathbf{k})) [r + (1-r) \cos^2 \theta(\mathbf{k})]}, \quad (4.17)$$

where $s = 0.25$ accounts for the spatial summation effect [PAW93a]. The parameters a_{MTF} , b_{MTF} , c_{MTF} , and d_{MTF} of the MTF are chosen as 2.46, 0.1, 0.25, and 1, respectively, as proposed by Chitprasert and Rao [CR90].

4.1.5 Integrating SBDCT and JND Model

As most images are spatially non-stationary, the perceptually significant subbands vary from sub-image to sub-image. Assuming $\mathbf{c}_{0,n}$ is the perceptually significant subband of the sub-images in an image, Jung *et al.* [JMM96] approximated the DCT-II using ApD1. Their approach significantly reduces the computational cost of SBDCT and works reasonably well, except at the edges of an image. Hence, their approach may not be suitable if transparent or near transparent coding of images is desired.

Figure 4.5 presents the images processed by DCT-II, approximated SBDCT (ApD1) and PSBDCT followed by reconstruction using inverse DCT-II (IDCT-II). To observe the distortion introduced by the various transforms, all the transform coefficients are not quantized. From the reconstructed images in Fig. 4.5, it is found that only the reconstructed image from ApD1 exhibits distortion. To understand the distortion found in the edges of Fig. 4.5(b), we can analyze the frequency response of the 2-D filters that are used to generate the subbands $\mathbf{c}_{b_i,n}$ from $\mathbf{c}_n$.

In this section, we investigate two 1-D filters that are used to generate $c(0, q_1, n_1)$ and $c(1, q_1, n_1)$ from $c(p_1, n_1)$, and this analysis can be easily

Fig. 4.5 Segments of the “Lena” image extracted from the constructed image, which are compressed using (a) DCT-II, (b) ApD1, and (c) PSBDCT and then reconstructed by IDCT-II decoder.

extended to the 2-D case. Using the flow diagram in Fig. 4.6, we arrive to the following:

$$\begin{aligned} a(m, n_1) &= \sum_{q_1=0}^1 h_a(q_1 - m) c(p_1, n_1), \\ b(m, n_1) &= \sum_{q_1=0}^1 h_b(q_1 - m) c(p_1, n_1), \end{aligned} \quad (4.18)$$

where $h_a(m)$ and $h_b(m)$ are 2-tap lowpass and highpass filters, respectively. The frequency response of these filters is shown in Fig. 4.7. Since these filters have poor stopband attenuation, significant aliasing is expected in $c(0, q_1, n_1)$ and $c(1, q_1, n_1)$. If $c(p_1, n_1)$ is of high frequency, approximating $c(p_1, n_1)$ using $c(0, q_1, n_1)$ can lead to significant distortion. In other words, approximating

Fig. 4.6 Signal flow diagram to decompose signal $c(p_1, n_1)$ into two sub-sequences $c(0, q_1, n_1)$ and $c(1, q_1, n_1)$.

Fig. 4.7 Frequency response of $h_a(m)$ and $h_b(m)$.

$c(p_1, n_1)$ using $c(1, q_1, n_1)$ can lead to visible distortion if $c(p_1, n_1)$ is of low frequency. This clearly suggests that $c(0, q_1, n_1)$ is not suited to approximate $c(p_1, n_1)$ for all cases.

In [CL95], Chou and Li used a 2-D quadrature mirror filterbank (QMF) [Joh80] to divide an image into 16 subbands. Subsequently, the authors employed 16 subband JND profiles to determine the perceptually significant components in

the 16 subbands. Likewise, we consider the decomposition stage in SBDCT as a filterbank that divides an $N \times N$ sub-image into M^2 subbands, where $M = 2$. Hence, M^2 subband JND profiles are used to determine the perceptual importance of these M^2 subbands.

For the proposed PIC, the perceptually insignificant subbands are removed from each $N \times N$ sub-image to achieve high compression ratio while preserving image quality. The remaining subbands (perceptually significant) in each $N \times N$ sub-image are transformed into the frequency domain by SBDCT, hence, this algorithm is referred as PSBDCT. To locate the perceptually insignificant subbands of an $N \times N$ sub-image, we compare the energy of the subbands of an $N \times N$ sub-image with the energy of its subband JND profiles. Let $en_{\text{sub}}(b_i, \mathbf{n})$ denote the energy of $C(b_i, \mathbf{k}, \mathbf{n})$ and $en_{\text{jnd}}(b_i, \mathbf{n})$ denote the energy of a $N/M \times N/M$ block at $(Nn_1/M, Nn_2/M)$ in the b_i th subband JND profile. $en_{\text{sub}}(b_i, \mathbf{n})$ and $en_{\text{jnd}}(b_i, \mathbf{n})$ are expressed as

$$\begin{aligned} en_{\text{sub}}(b_i, \mathbf{n}) &= \sum_{m_1=0}^{N/M-1} \sum_{m_2=0}^{N/M-1} c^2(b_i, \mathbf{m}, \mathbf{n}), \\ en_{\text{jnd}}(b_i, \mathbf{n}) &= \sum_{m_1=0}^{N/M-1} \sum_{m_2=0}^{N/M-1} t_{\text{NMM}, M^2}^2 \left(b_i, \frac{Nn_1}{M} + m_1, \frac{Nn_2}{M} + m_2 \right), \end{aligned} \quad (4.19)$$

where t_{NMM, M^2} is the M^2 subband JND profiles computed by substituting (4.1) into (2.42). By comparing the energy of the b_i th subband in each $N \times N$ sub-image and its corresponding $N/M \times N/M$ block of the b_i th subband JND, we have

$$pis(b_i, \mathbf{n}) = \begin{cases} 1, & \text{for } en_{\text{sub}}(b_i, \mathbf{n}) > en_{\text{jnd}}(b_i, \mathbf{n}), \\ 0, & \text{otherwise,} \end{cases} \quad (4.20)$$

where $pis(k_s, \mathbf{n})$ denotes the location of the perceptually important subbands in each $N \times N$ sub-image. From (3.7), PSBDCT is computed by

$$C_{\text{pct}}(\mathbf{k}, \mathbf{n}) = \sum_{b_i=0}^{M^2-1} pis(b_i, \mathbf{n}) C(b_i, \mathbf{k}, \mathbf{n}). \quad (4.21)$$

To prevent over estimation of the JND threshold at the sub-images, which containing removed subbands (perceptually insignificant subbands), the values of the JND profile at sub-images are reduced. Let $t_{\text{upd}}(\mathbf{x})$ denote the updated JND profile. The updating of the JND profile which accounts for the removal of perceptually insignificant subbands of the image is given as

$$t_{\text{upd}}(\mathbf{x}) = t_{\text{NAMM}}(\mathbf{x}) \sum_{b_i=0}^{M^2-1} \left[1 - pis\left(b_i, \left\lfloor \frac{x_1}{N} \right\rfloor, \left\lfloor \frac{x_2}{N} \right\rfloor\right) w_{M^2}(b_i) \right], \quad (4.22)$$

therefore, the updated M^2 subband JND profile $t_{\text{upd}, M^2}(b_i, \mathbf{s})$ becomes

$$t_{\text{upd}, M^2}^2(b_i, \mathbf{s}) = t_{\text{NAMM}, M^2}^2(b_i, \mathbf{s}) \cdot \left[\sum_{b_i=0}^{M^2-1} \left(1 - pis\left(b_i, \left\lfloor \frac{Ms_1}{N} \right\rfloor, \left\lfloor \frac{Ms_2}{N} \right\rfloor\right) w_{M^2}(b_i) \right) \right]^2. \quad (4.23)$$

As discussed by Chitprasert and Rao [CR90], the base detection threshold derived from any MTF should be adjusted for every $N \times N$ sub-image to adapt to the local activities of an image. To perform this local adaptation of the base detection threshold, M^2 and N^2 subband JND profiles of an image is used. Specifically, the proposed PIC uses M^2 subband JND profiles to determine the perceptually insignificant subbands of each sub-image, and then uses N^2 subband JND profiles to locally adapt the quantization matrix.

To reduce computational cost, we directly compute N^2 subband JND profiles $t_{N^2}(b_q, \mathbf{n})$ from the M^2 subband JND profiles, where $b_q = 0, 1, \dots, N^2 - 1$. From

(2.42), it is clear that $t_{N^2}(b_q, \mathbf{n})$ can be computed by summing $N/M \times N/M$ blocks of $t_{M^2}(b_i, \mathbf{s})$. With this in mind, we establish the following:

$$t_{N^2}^2(b_q, \mathbf{n}) = \frac{w_{N^2}(b_q)}{w_{M^2}(0)} \sum_{j_1=0}^{N/M-1} \sum_{j_2=0}^{N/M-1} t_{\text{upd}, M^2}^2\left(0, \frac{N}{M}n_1 + j_1, \frac{N}{M}n_2 + j_2\right). \quad (4.24)$$

Finally, we compute the locally adapted quantization matrix using

$$T_{\text{prop}}(\mathbf{k}, \mathbf{n}) = \sqrt{T_{b, \text{MTF}}(\mathbf{k}) t_{N^2}^2(Nk_1 + k_2, \mathbf{n})}. \quad (4.25)$$

4.2 Verification of JND Modeling

Chou and Li [CL95] verified the performance of their JND model by studying its effectiveness of noise shaping for images. One of the primary objectives of integrating a JND model with a PIC is to minimize introducing perceptible noise in the image compression progress. For a pixel-based JND model $t(\mathbf{x})$, a noise-contaminated image can be obtained as

$$c_{nc}(\mathbf{x}) = c(\mathbf{x}) + \tau \text{rand}(\mathbf{x}) t(\mathbf{x}), \quad (4.26)$$

where $\text{rand}(\mathbf{x})$ takes on values of 1 or -1 randomly; $c(\mathbf{x})$ and $c_{nc}(\mathbf{x})$ denote the original and noise-contaminated images, respectively; τ is a scaling factor larger than one to control the amount of error energy injected into the image.

This section compares the JND models from [Wat93, CL95, YLL05, ZLX05, WN09], and to validate the JND model adopted in the proposed PIC. The JND models from [Wat93, CL95, YLL05, ZLX05, WN09] are referred as t_{Wat} , t_{CL} , t_{YLL} , t_{ZLX} , and t_{WN} , respectively, and the proposed JND model is referred as t_{prop} . Among these JND models, only t_{CL} and t_{YLL} are pixel-based JND models. The rest of the JND models are DCT subband-based JND models, except for the

proposed JND model that is obtained with a unique combination of pixel and subband JND models.

A direct comparison of the six JND models is not possible. In [ZLX08], Zhang *et al.* proposed a method which facilitates the comparison between subband-based JND models and pixel-based JND models. Their method involves estimating JND values of the pixels at the n th sub-image by considering the contribution of DCT subbands at the n th DCT block. Their method starts with computing new set of JND threshold $T'(\mathbf{k}, \mathbf{n})$ by discarding any DCT coefficient lower than JND threshold $T(\mathbf{k}, \mathbf{n})$. This new set of threshold is computed as

$$T'(\mathbf{k}, \mathbf{n}) = \begin{cases} \text{sign}(C(\mathbf{k}, \mathbf{n}))T(\mathbf{k}, \mathbf{n}), & \text{for } |C(\mathbf{k}, \mathbf{n})| \geq t(\mathbf{k}, \mathbf{n}), \\ 0, & \text{otherwise,} \end{cases} \quad (4.27)$$

where $\text{sign}(C(\mathbf{k}, \mathbf{n}))$ produces the sign of $C(\mathbf{k}, \mathbf{n})$. Subsequently, the compound effect of the DCT subbands is obtained by summing all the DCT subbands of the n th DCT block using (2.43). Let $t'(\mathbf{x})$ denote the estimated pixel-based threshold from the DCT subbands computed by (2.43). Using NAMM, a pixel-based JND model $t(\mathbf{x})$ is estimated as

$$t(\mathbf{x}) = t'_{la}(\mathbf{x}) + t'(\mathbf{x}) - C_{ol} \times \min\{t'_{la}(\mathbf{x}), t'(\mathbf{x})\}, \quad (4.28)$$

where

$$t'_{la}(\mathbf{x}) = \frac{T\left(0, 0, \left\lfloor \frac{x_1}{N} \right\rfloor, \left\lfloor \frac{x_2}{N} \right\rfloor\right)}{N^2}. \quad (4.29)$$

Using (4.27) to (4.29), the pixel-domain JND models of [Wat93, CL95, YLL05, ZLX05, WN09] and the proposed JND model are estimated.

The comparison of the six JND models is carried out by subjective evaluation. Ten subjects made up of two females and eight males, with age between 20 to 36

years old, are chosen in this subjective evaluation. The subjective evaluation of the six JND models are performed with 20 test-images TI-1 to TI-20 of 512×512 pixels. The monitor used in this test is the Acer P205H LCD monitor, which has a native resolution of 1600×900 pixels. The viewable screen of this monitor is 442.800 mm by 249.075 mm, hence the width and the height of a pixel are 0.277 mm. The viewing distance (six times of image height) at this pixel pitch is 850 mm. The subjective evaluation was conducted in a room, which is illuminated by fluorescent lights. At any one time, two images were displayed on the screen where the left as the original image and the right image is the processed or noise-contaminated image. The subjects were asked to record his or her assessment of each noise contaminated image as compared to the original image according to Table 4.2.

Table 4.2 Description of subjective rating used in subjective evaluation.

Subjective Score	Description
0	Right image is the same quality as the left image
1	Right image is of very good quality as compared to left image, but some differences can be observed
2	Right image is worse than left image
3	Right image is much worse than left image

Zhang *et al.* [ZLX08] observed that pixel-based JND models tend to be inferior as these JND models do not take into account of the contrast sensitivity of the HVS. Adapting the approach used in [LMK98], Zhang *et al.* used two vertical gratings to evaluate the ability of the six JND models in discerning noise (grating injected into test-images as noise) at different frequencies. In this section, we used two horizontal gratings since the viewing distance of the test-images depends on

the image height. The grating frequencies used in our subjective evaluation are 4 and 20 cpd. Since the horizontal grating having 4 cpd falls under the sensitive range of the HVS, it is expected to be highly visible, whereas the horizontal grating at 20 cpd should be barely visible.

Six JND profiles of TI-6 having horizontal grating at 4 and 20 cpd are shown in Fig. 4.8 and 4.9, respectively. The bright and dark regions of a JND profile indicate high and low JND threshold, respectively. From the six JND profiles of TI-6 in Fig 4.8 and 4.9, it is clear that the pixel-based JND models are unable to detect the highly visible grating at 4 cpd. In fact, the two horizontal gratings are not detected by the pixel-based models. On the other hand, all the DCT subband-based JND models detect the horizontal grating at 4 cpd, which is consistent with subjective viewing. This observation reinforces the conclusion in [ZLX08] that pixel-based JND models are incapable of accurately detecting periodic noise.

From Fig. 4.8 and 4.9, it is evident that the different block classification techniques used in t_{Wat} , t_{ZLX} , t_{WN} , and t_{prop} result in significantly different JND profiles of the same test-image. The JND profile of TI-6 based on t_{Wat} exhibits block-like characteristic, especially at the shoulder area of "Lena" image. Therefore, Watson's JND model can lead to perceptible artifacts at the edge and texture regions, which is due to abrupt changes of JND threshold resulting in block-like characteristic at these regions. Similarly, abrupt changes of JND values with block-like characteristic are found at edge and texture regions of the JND profile based on t_{ZLX} , as shown in Fig. 4.8(e) and 4.9(e).

(a)

(b)

(d)

(e)

(f)

(g)

Fig. 4.8 JND profiles of TI- 6 with horizontal grating at 4 cpd. JND profiles based on t_{Wat} , t_{CL} , t_{YLL} , t_{ZLX} , t_{WN} , and t_{prop} are shown in (a) to (g), respectively.

(a)

(b)

(d)

(e)

(f)

(g)

Fig. 4.9 JND profiles of TI-6 with horizontal grating at 20 cpd. JND profiles based on t_{Wat} , t_{CL} , t_{YLL} , t_{ZLX} , t_{WN} , and t_{prop} are shown in (a) to (g), respectively.

Changes in the JND profiles of TI-6 based on t_{WN} and t_{prop} do not exhibit the block-like characteristic like the JND profiles based on t_{Wat} and t_{ZLX} . However, numerous small bright areas at the feather and hair regions are found in Fig. 4.8(f) and 4.9(f). It is also clear that the JND profiles based on t_{WN} and t_{prop} are comparably lower than those based on t_{ZLX} . On the other hand, the JND profile based on t_{prop} is found to have a more consistent JND threshold at the hat and hair regions of TI-6. It is generally preferred to have consistently high JND threshold at the feather and hair regions since these regions are potentially good maskers of noise. This observation is also found in the JND profiles based on the pixel-based JND models t_{CL} and t_{YLL} .

In some cases, the two horizontal gratings are invisible to the HVS due to contrast masking. This is generally observed in images containing significant amount of high spatial frequency content (such as TI-10). In such test-images, a good JND model is expected not to detect the horizontal grating at 4 and 20 cpd, if contrast masking occurs. The JND profiles of TI-10 with horizontal gratings at 4 and 20 cpd are shown in Fig. 4.10. For each JND model, the JND profile of TI-10 with the horizontal grating of 4 and 20 cpd are shown on the left and right, respectively. From Fig. 4.10, it is clear that all the JND models do not detect any horizontal grating.

To compare the noise shaping performance of the six JND models, noise is injected into 20 test-images (TI-1 to TI-20) to produce noise-contaminated images [CL95, YLL05, ZLX08]. Using (4.26), noise is injected into the test-images based on the six JND models. The scaling factor τ in (4.26) is adjusted to produce noise-contaminated images having PSNR of 30 dB. The subjective rating for the noise-

Fig. 4.10 JND profiles of TI-10 with horizontal grating at 4 (left image) and 20 (right image) cpd. JND profiles based on t_{Wat} , t_{CL} , t_{YLL} , t_{ZLX} , t_{WN} , and t_{prop} are shown in (a) to (g), respectively.

contaminated images is summarized in Table 4.3, and the 95% confidence interval of the subjective rating is shown in Fig. 4.11. t_{CL} and t_{prop} produce the worst and best subjective rating among the six JND models, respectively. It is evident that t_{YLL} outperforms t_{CL} in terms of JND modeling. This observation supports our decision to adopt NAMM to estimate the pixel-based JND from DCT subbands, instead of the maximum operator suggested by Chou and Li [CL95]. Table 4.3 and Fig. 4.11 also reveal that the DCT subband-based JND models t_{Wat} , t_{ZLX} , and t_{WN} perform similarly, and t_{WN} being marginally better than the rest of the DCT subband-based JND models. The DCT subband-based and proposed JND models (t_{Wat} , t_{ZLX} , t_{WN} , and t_{prop}) are found to be superior than pixel-based JND models. Again, this observation is consistent with the results reported in [ZLX08]. The inferior performance of the pixel-based JND models can be explained with the omission of CSF in their modeling of the HVS.

It is interesting to note that all the pixel-based JND models outperform the DCT subband-based and proposed JND models in the case of TI-8. A visual inspection of TI-8 reveals that this test-image, as shown in Fig. 4.12(a), contains significant amount of grain noise. From Fig. 4.12, it is clear that the noise-contaminated images from DCT subband-based JND models are comparably worse than those obtained using pixel-based JND models. Specifically, noise-contaminated images from DCT subband-based JND models exhibit more noise in the background, hair and eyes regions of TI-8 as compared to the noise-contaminated images obtained with pixel-based JND models. Among the DCT subband-based and proposed JND models, it is clear that the noise-contaminated image from t_{Wat} contains the most perceptible distortion, especially at the edges of

Table 4.3 Subjective rating for noise-contaminated images (PSNR = 30 dB) from various JND models.

Test Image	Subjective Rating (0 – 3)					
	t_{Wat}	t_{CL}	t_{YLL}	t_{ZLX}	t_{WN}	t_{prop}
TI-1	1.5	1.8	1.4	1.2	1.5	0.8
TI-2	1.8	1.8	1.6	1.8	1.6	1.7
TI-3	1.3	1.2	1.2	1.3	1.4	1.3
TI-4	1.4	2.6	2.1	1.3	1.2	1.1
TI-5	1.5	2.4	1.4	1.4	1.4	1.3
TI-6	1.5	1.8	1.7	1.6	1.5	1.3
TI-7	1.7	2.3	1.8	1.7	1.7	1.5
TI-8	2.2	1.7	1.7	2.1	2.1	1.9
TI-9	1.6	1.9	1.5	1.7	1.8	1.5
TI-10	0.8	1.3	1.0	1.1	1.1	0.6
TI-11	0.9	1.3	0.8	0.9	0.8	0.6
TI-12	1.5	1.6	1.7	1.6	1.5	1.2
TI-13	1.2	1.2	1.1	1.3	1.2	1.1
TI-14	1.5	1.4	1.4	1.3	1.2	1.0
TI-15	1.5	1.8	1.6	1.4	1.5	1.3
TI-16	1.7	1.8	2.0	1.6	1.5	1.4
TI-17	1.4	2.0	1.8	1.2	1.2	0.8
TI-18	0.6	0.6	0.8	0.6	0.7	0.4
TI-19	2.0	2.3	2.1	2.0	1.9	2.3
TI-20	2.0	1.8	2.1	1.8	1.9	1.6
Mean Rating	1.48	1.73	1.54	1.445	1.435	1.235

Fig. 4.11 Subjective rating and its associated 95% confidence interval of noise-contaminated images (PSNR = 30 dB) from various JND models.

(a)

(b)

(c)

(d)

(e)

(f)

(g)

Fig. 4.12 Noise-contaminated images of TI-8 from various JND models, where (a) is the cropped original image. Noise-contaminated images from t_{Wat} , t_{CL} , t_{YLL} , t_{ZLX} , t_{WN} , and t_{prop} are (b) to (g), respectively.

the hair near the forehead and around the eyes. t_{Wat} and t_{ZLX} inject significant amount of noise into the background of the TI-8, thus, noise is also more noticeable at the background of the noise-contaminated images obtained with t_{Wat} and t_{ZLX} as compared to those based on t_{WN} and t_{prop} . Since the estimated pixel-based JND models of the DCT subband-based and proposed JND models are obtained by summing all the JND thresholds within an $N \times N$ block, DCT subband-based and proposed JND models are incapable of accurately estimating the JND threshold for changes in a small region of an image, such as in the case of grain noise.

Among the three DCT subband-based JND models, t_{Wat} is the only JND model that does not perform block classification, which leads to a poorer estimation of contrast masking as compared to t_{ZLX} , t_{WN} , and t_{prop} . It is observed that this JND model results in visible distortion at edges, especially at bright regions of an image. This can be attributed to the inaccurate estimation of luminance adaptation and contrast masking. This distortion is prominent in several test images, especially TI-17 and TI-20. To highlight this observation, the cropped images of TI-17 and TI-20 are shown in Fig. 4.13. Among the noise-contaminated images, highly perceptible distortion is found at the two leftmost windows and the vertical stabilizer of the aircraft tail in Fig. 4.13(c) and 4.13(d), respectively.

Visible distortion is also seen on the aircraft body in Fig. 4.13(d). From Fig. 4.13, t_{prop} produces the least perceptible distortion at the edges of the two leftmost windows for TI-17. Similar observation is made from the noise-contaminated images of TI-20. Comparing Fig. 4.13(g) and 4.13(i), t_{prop} produces visually better

edges as compared to t_{WN} , but t_{WN} exhibits a slight improvement over t_{prop} at the edges at the aircraft tail of TI-20, as shown in Fig. 4.13(h) and 4.13(j).

Fig. 4.13 Noise-contaminated images obtained from DCT subband-based JND models. (a) Original TI-17, (b) original TI-20, (c) to (d) Watson's model, (e) to (f) Zhang's model, (g) to (h) Wei and Ngan's model, and (i) to (j) proposed model.

4.2.1 Error Pooling

For many image coders, the input image is processed by a filterbank, where each filter in the filterbank possesses different spatial frequency. Assuming the images from each filter are lossy coded, the distortion of each filtered image can be presented as a distortion map. To obtain a single distortion map describing the reconstructed image from the filterbank, one commonly used approach is to combine all the distortion maps across frequency, spatially, or both, using the Minkowski metric.

At near JND threshold, probability summation is well accepted as the basis for summation of signal energy (or distortion) across frequency and spatial domains [EB98]. For summation across frequency band, it is found that $\beta = 3.5$ [Wat82] is consistent with subjective evaluation reported in [GRN78, Leg78a, Leg78b, RG81]. It has been found that summation across frequency band with DCT basis functions is well modeled with a parameter of $\beta = 2.4$ [PAW93b]. In target detection experiments [RAW97], it is found that $\beta = 4$ provides the closest match to psychophysical results for spatial summing.

In Chapter 2, error pooling across frequency band is performed using (2.38). To obtain a single distortion value P describing the amount of distortion in a compressed image, spatial summing is performed after summation across frequency bands or vice versa. The perceptual distortion score P is computed as

$$P = \left(\sum_{\mathbf{n}} P(\mathbf{n})^\beta \right)^{1/\beta}. \quad (4.30)$$

To adapt to the local activities of an image, a quantization matrix is defined for every $N \times N$ sub-image, which increases the bit-rate of a compressed image. To avoid this problem, Watson [Wat93] derived a global quantization matrix

according to the JND threshold of an image. By defining all of $P(\mathbf{k})$ (Watson consider spatial summation first then summation across frequency bands) to be lesser than a pre-defined distortion P_{def} , a quantization matrix leading to the lowest bit-rate at P_{def} is obtained.

Various values of β have been proposed in JND models. Zhang *et al.* [ZLX05, ZLX08] suggested the following for summing across frequency band:

$$P(\mathbf{n}) = \begin{cases} \left(\sum_{\mathbf{k}} |d_{\text{JND}}(\mathbf{k}, \mathbf{n})|^{2.3} \right)^{1/2.3}, & \text{for } \mathbf{k} = (0,0), (1,0), (0,1), \\ \left(\sum_{\mathbf{k}} |d_{\text{JND}}(\mathbf{k}, \mathbf{n})|^4 \right)^{1/4}, & \text{otherwise,} \end{cases} \quad (4.31)$$

where $d_{\text{JND}}(\mathbf{k}, \mathbf{n}) = (C(\mathbf{k}, \mathbf{n}) - \hat{C}(\mathbf{k}, \mathbf{n})) / t(\mathbf{k}, \mathbf{n})$, and $\hat{C}(\mathbf{k}, \mathbf{n})$ is the quantized $N \times N$ DCT coefficients of the $\mathbf{n}$ th sub-image. Similar to t_{Wat} , the spatial summing is performed with $\beta = 4$.

To determine a suitable value of β for the proposed JND model, three sets of β for spatial and frequency summation are compared in the next subjective evaluation. Since different values of β can be used for spatial and frequency summation, we denote $\boldsymbol{\beta} = \{\beta_s, \beta_f\}$, where β_s and β_f are the values of β defined for spatial and frequency summation, respectively. Three sets of $\boldsymbol{\beta}$ is used in this evaluation and are as referred as $\boldsymbol{\beta}_1$, $\boldsymbol{\beta}_2$, and $\boldsymbol{\beta}_3$. Based on the observations from [PAW93b, RAW97], $\boldsymbol{\beta}_1$ is chosen to be $\{4, 2.4\}$. Referenced from [ZLX05, ZLX08], frequency summation is carried out using (4.31) with $\boldsymbol{\beta}_2$, and a value of four is used for spatial summation. Referenced from [Wat93], $\boldsymbol{\beta}_3 = \{4, 4\}$ is used for spatial and frequency summation.

To determine the suitable values of β for spatial and frequency summation, subjective evaluation is conducted with compressed images at 0.3 to 0.5 bpp (bit per pixel). These compressed images are obtained with the proposed JND model using the quantization matrix defined in (4.25) together with error pooling given by (4.30) and (4.31). Table 4.4 summaries the average subjective rating for 10 subjects in the subjective evaluation of the compressed images obtained with the proposed JND model, and the 95% confidence interval of the subjective rating is shown in Fig. 4.14. The following observations are made from Table 4.4 and Fig. 4.14. As expected, the mean of the subjective rating reduces as bit-rate increases. For all bit-rates, it is clear that β_2 and β_3 produces compressed images having the best and worst image quality among the three sets of β , respectively. In addition, β_2 leads to the lowest subjective rating with the smallest confidence interval of the subjective rating among the three sets of β . The ten subjects observed that the compressed images obtained using β_3 tends to exhibit severe blocking artifacts as compared to the compressed images obtained using β_1 and β_2 , especially at 0.3 bpp. These observations suggest that blocking artifact can be reduced by using a smaller value of β_s , which is used in β_1 and β_2 . Furthermore, our subjective evaluation confirms that a smaller value of β_f to sum across the lower spatial frequency bands of $d_{\text{JND}}(\mathbf{k}, \mathbf{n})$ yield compressed images with better image quality.

Ideally, the perceptual distortion score P computed from a JND model after pooling error correlates with the subjective rating r of an image. The relation between P and r of an image may be non-linear as subjective evaluation can have non-linear quality rating compression at the extremes of subjective rating. To

Table 4.4 Subjective rating for compressed images obtained with proposed JND model using various values of β in error pooling.

Test Image	Rating for 0.3 bpp			Rating for 0.4 bpp			Rating for 0.5 bpp		
	β_1	β_2	β_3	β_1	β_2	β_3	β_1	β_2	β_3
TI-1	2.8	3.0	2.9	2.7	2.5	2.4	2.1	1.5	1.9
TI-2	2.3	2.1	2.2	1.6	1.4	1.3	1.3	1.0	1.6
TI-3	2.1	1.8	2.2	1.5	1.3	1.7	1.1	0.8	1.3
TI-4	2.2	1.6	2.3	1.8	1.3	1.7	1.3	0.8	1.2
TI-5	2.7	2.3	2.9	2.0	1.0	2.3	1.2	0.9	1.5
TI-6	1.6	1.7	1.5	1.2	1.2	1.3	1.0	0.9	1.0
TI-7	1.7	1.4	1.5	0.5	0.6	0.4	0.3	0.3	0.4
TI-8	1.0	1.2	1.4	0.4	0.3	0.3	0.3	0.4	0.3
TI-9	1.8	1.4	1.6	1.5	1.2	1.6	1.3	0.9	1.4
TI-10	2.1	1.8	2.3	1.7	1.4	1.9	1.2	1.1	1.4
TI-11	2.6	2.2	2.8	2.1	1.3	2.3	1.5	1.0	1.8
TI-12	2.5	1.9	2.4	1.8	1.3	1.9	1.6	0.8	1.7
TI-13	2.9	2.6	2.8	2.7	1.9	2.8	2.2	1.6	2.3
TI-14	2.2	1.8	2.3	2.0	1.3	2.1	1.5	1.0	1.7
TI-15	2.5	2.2	2.7	2.3	1.9	2.5	1.8	1.4	2.0
TI-16	1.9	2.1	2.3	1.4	1.3	1.5	1.0	1.1	1.2
TI-17	2.3	2.2	2.4	1.6	1.5	1.4	1.2	1.1	1.2
TI-18	2.1	1.7	2.0	1.9	1.5	2.0	1.6	1.2	1.7
TI-19	1.7	1.6	1.9	1.6	1.2	1.7	1.3	0.9	1.2
TI-20	2.1	1.9	2.3	1.6	1.3	1.7	1.3	1.1	1.2
Mean Rating	2.16	1.93	2.24	1.69	1.34	1.74	1.31	0.99	1.40

Fig. 4.14 Subjective rating and its associated 95% confidence interval of noise-contaminated images at various bit-rates and values of β in error pooling of the proposed JND model.

minimize the non-linearity due to the subjective evaluation, and to facilitate comparison of various JND models, the relation between each JND model's distortion score and its subjective rating is estimated using non-linear regression [VQE03]. In this case, the rating $r'(P)$ is curve fitted using a four-parameter cubic polynomial

$$r'(P) = a_0 + a_1P + a_2P^2 + a_3P^3, \quad (4.32)$$

where a_0 , a_1 , a_2 , and a_3 are fitted to a P versus r curve. The video quality experts group (VQEG) employs the Pearson correlation ρ_p and Spearman rank-order correlation ρ_s to assess the prediction accuracy and prediction monotonicity of a distortion metric, respectively. In addition, the root mean square (RMS) error between $r'(P)$ and the actual subjective rating r is computed to measure the accuracy of $r'(P)$.

Pearson correlation and Spearman rank-order correlation are defined as follow:

$$\rho_p = \frac{\sum_k (r'(P_k) - \bar{r})(P_k - \bar{P})}{\sqrt{\sum_k (r'(P_k) - \bar{r})^2} \sqrt{\sum_k (P_k - \bar{P})^2}}, \quad (4.33)$$

and

$$\rho_s = \frac{\sum_k (\chi_k - \bar{\chi})(\gamma_k - \bar{\gamma})}{\sqrt{\sum_k (\chi_k - \bar{\chi})^2} \sqrt{\sum_k (\gamma_k - \bar{\gamma})^2}}, \quad (4.34)$$

where $r(P_k)$ is the predicted rating for test-image k and $\bar{r}$ is the mean of $r'(P_k)$; P_k is the perceptual distortion score of test-image k and $\bar{P}$ is the mean of P ; χ_k is the rank-ordered series of $r'(P_k)$ and $\bar{\chi}$ is the mid-rank of $r'(P_k)$; γ_k is the rank-

ordered series of P_k , and $\bar{\gamma}$ is the mid-rank of P_k . The Pearson correlation and Spearman rank-order correlation range between from -1 to 1, therefore, the ideal match between r' and P is $|\rho_p|, |\rho_s| = 1$. RMS error r_e is defined as

$$r_e = \frac{1}{N_k} \sum_k (r'(P_k) - r_k)^2, \quad (4.35)$$

where N_k is the number of test-images.

The scatter plots for subjective rating versus perceptual distortion score of the proposed JND model with β_1 , β_2 , and β_3 , and its associated 95% confidence interval is shown in Fig. 4.15 and 4.16, respectively. The rating $r'(P)$ is indicated by a red line in each scatter plot. All the scatter plots in Fig. 4.15 indicate a positive and almost linear (except for Fig. 4.15(a)) relation between the subjective rating and perceptual distortion score. This is highly desirable as the perceptual distortion score is to be used as an indicator of the perceived quality of a compressed image. However, the scatter plots in Fig. 4.15(a) and 4.15(c) exhibit significant spreading of the points in the plots, which suggest significant deviation between predicted distortion score and subjective rating. Also, it is evident from Fig. 4.15 that the perceptual distortion score computed with β_1 and β_3 do not correlate with subjective rating as well as the perceptual distortion score obtained with β_2 .

The values of ρ_p , ρ_s , and r_e of proposed JND model with β_1 , β_2 , and β_3 are tabulated in Table 4.5, and the confidence interval of ρ_p and ρ_s is shown in Fig. 4.16. Coinciding with our earlier observations made from Fig. 4.15, the highest Pearson correlation and Spearman correlation in Fig. 4.16 are associated with β_2 . On the other hand, the confidence interval of the subjective rating associated with

(a)

(b)

(c)

Fig. 4.15 Scatter plots of subjective rating versus perceptual distortion score. Fitted polynomial function is shown in red. Scatter plots of proposed PIC using β_1 , β_2 , and β_3 are shown in (a), (b), and (c), respectively.

Table 4.5 Performance measures of proposed JND using various β used error pooling.

Values used for β	Performance Measure		
	ρ_p	ρ_s	r_e
β_1	0.766	0.771	0.149
β_2	0.858	0.849	0.105
β_3	0.709	0.686	0.169

Fig. 4.16 Subjective rating and its associated 95% confidence interval of noise-contaminated images for various values of β in error pooling of the proposed JND model.

β_3 is found to be the largest. Based on these observations, the values defined in β_2 is selected for the error pooling in the proposed JND model.

4.3 Performance Comparison of PICs

In this section, we compare the compression performance of four PICs having DCT subband-based JND models (t_{Wat} , t_{ZLX} , and t_{WN}) and the proposed JND model t_{prop} . Since the Wei and Ngan [WN09] didn't define an error pooling stage

for t_{WN} , the error pooling algorithm used in t_{prop} is adopted in t_{WN} to facilitate this comparison.

The luminance adaptation used in t_{Wat} results in $P(\mathbf{n}) = \infty$ for some DCT blocks. This problem is observed with TI-3, 8, 10, 11, and 13, and is caused by the elevation parameter $e_1^{\text{Wat}}(\mathbf{n}) = 0$ when the DC coefficient of a DCT block is zero. This leads to JND threshold $t(\mathbf{k}, \mathbf{n}) = 0$, and causing $P(\mathbf{n})$ to go infinity. To avoid $P(\mathbf{n})$ to go infinity, the elevation parameter $e_1^{\text{Wat}}(\mathbf{n})$ is modified to

$$e_1^{\text{Wat}}(\mathbf{n}) = \begin{cases} \left(\frac{1}{C_{L_a}} \right)^{\alpha^T}, & \text{for } C(0, 0, \mathbf{n}) = 0, \\ \left(\frac{C(0, 0, \mathbf{n})}{C_{L_a}} \right)^{\alpha^T}, & \text{otherwise.} \end{cases} \quad (4.36)$$

As in the previous section, the quality of the compressed images at 0.3 to 0.5 bpp is evaluated by 10 subjects, and the subjective rating for the 20 test-images is summarized in Table 4.6, and the confidence interval of the subjective rating is shown in Fig. 4.17. From Table 4.6 and Fig. 4.17, compressed images from the proposed PIC are found to have the lowest subjective rating at 0.4 and 0.5 bpp. This observation suggest that the compressed images from the proposed PIC are visually better than those from other PICs at these bit-rates. At 0.3 bpp, all the PICs performed similarly except Watson's PIC, and the average subjective rating ranges from 1.89 (Zhang) to 2.17 (Watson). It is found that Zhang's PIC performs marginally better than the proposed PIC at 0.3 bpp.

Watson's PIC [Wat93] is consistently found to the worst performer for all bit-rates, and compressed images obtained with Watson's PIC tend to exhibit the most blocking artifacts among the four PICs. However, it is found that several

Table 4.6 Subjective rating for compressed images from various PICs.

Test Image	Rating for 0.3 bpp				Rating for 0.4 bpp				Rating for 0.5 bpp			
	Wat	ZLX	WN	Prop	Wat	ZLX	WN	Prop	Wat	ZLX	WN	Prop
TI-1	2.9	3.0	2.6	3.0	2.7	2.8	2.6	2.5	2.1	1.7	2.0	1.5
TI-2	2.5	2.1	2.3	2.1	2.1	1.6	1.8	1.4	1.6	1.1	1.5	1.0
TI-3	2.3	1.7	2.1	1.8	2.1	1.2	1.5	1.3	1.5	0.9	1.2	0.8
TI-4	2.1	1.5	1.8	1.6	1.8	1.3	1.5	1.3	1.2	1.0	1.3	0.8
TI-5	2.5	2.0	2.2	2.3	1.3	0.9	1.2	1.0	1.0	0.8	1.0	0.9
TI-6	2.2	1.5	1.8	1.7	1.6	1.2	1.4	1.2	1.1	0.7	0.9	0.9
TI-7	1.3	1.1	1.5	1.4	0.7	0.5	0.8	0.6	0.5	0.4	0.6	0.3
TI-8	1.5	1.2	1.1	1.2	0.2	0.2	0.4	0.3	0.3	0.3	0.3	0.4
TI-9	1.6	1.3	1.4	1.4	1.2	1.1	1.3	1.2	0.8	0.9	1.0	0.9
TI-10	2.3	1.7	2.1	1.8	1.8	1.5	1.5	1.4	1.2	1.3	1.2	1.1
TI-11	2.5	2.3	2.0	2.2	2.2	2.1	1.4	1.3	1.6	1.2	1.4	1.0
TI-12	2.3	2.0	1.7	1.9	1.5	1.2	1.2	1.3	1.1	0.9	1.0	0.8
TI-13	3.0	3.0	2.4	2.6	2.4	2.5	2.0	1.9	2.1	1.5	1.5	1.6
TI-14	1.6	1.9	2.0	1.8	1.3	1.4	1.5	1.3	1.1	1.1	1.4	1.0
TI-15	2.6	2.7	2.5	2.2	2.4	2.3	2.0	1.9	1.7	1.5	1.6	1.4
TI-16	1.9	1.7	2.2	2.1	1.5	1.5	1.7	1.3	1.3	1.2	1.7	1.1
TI-17	1.9	2.1	2.0	2.2	1.6	1.4	1.6	1.5	1.1	1.0	1.3	1.1
TI-18	2.1	1.7	1.8	1.7	1.8	1.6	1.6	1.5	1.5	1.1	1.4	1.2
TI-19	2.2	1.6	1.7	1.6	1.6	1.3	1.4	1.2	1.2	0.9	1.0	0.9
TI-20	2.0	1.7	2.0	1.9	1.5	1.5	1.7	1.3	1.3	1.1	1.4	1.1
Mean Rating	2.17	1.89	1.96	1.93	1.67	1.46	1.51	1.34	1.27	1.03	1.24	0.99

Fig. 4.17 Subjective rating and its associated 95% confidence interval of noise-contaminated images from various PICs.

compressed images obtained using Watson's PIC have significant sharper edges and textures as compared to the compressed images from other PICs. Generally, the image quality of these compressed images are degraded due to the significant blocking artifacts, except for TI-14. The observation of these test-images is summarized in Table 4.7. On the other hand, many compressed images obtained with Wei's PIC exhibit significant blurring artifacts. A list of test-images that exhibit more blurring artifacts with Wei's PIC than the rest of the PICs is tabulated in Table 4.8.

Table 4.7 Regions of compressed images from Watson's PIC having sharp edges.

Test Image	Bit-rate (bpp)	Region of image
TI-5	0.4 - 0.5	Tree and Roof top
TI-6	0.4 - 0.5	Hat
TI-14	0.3 - 0.5	Mountain
TI-15	0.3 - 0.5	Tree
TI-16	0.3 - 0.5	Shell
TI-18	0.3 - 0.5	Fur and eyes

Table 4.8 Regions of compressed images from Wei's PIC having blur edges.

Test Image	Bit-rate (bpp)	Region of image
TI-4	0.3 - 0.5	Boat mast
TI-5	0.5	Roof and tree
TI-9	0.3 - 0.5	Window and roof
TI-10	0.3 - 0.5	Tree and brick
TI-11	0.3 - 0.5	Fish scale
TI-14	0.3 - 0.5	Mountain
TI-15	0.3 - 0.5	Tree

The contrast masking $e_2^{\text{Wei}}(\mathbf{k}, \mathbf{n})$ of Wei's PIC is estimated slightly different from Watson's PIC as Watson did not use block classification in his contrast masking estimation. Zhang's and Wei's PICs adopted a similar strategy to estimate contrast masking for an edge block, which assigns higher contrast masking for DCT coefficients having higher spatial frequencies. Comparing (2.32) with

$$e_2^{\text{Wei}}(\mathbf{k}, \mathbf{n}) = \begin{cases} ec_{\text{intra}}^{\text{Wei}}(\mathbf{n}), & \text{for } k_1^2 + k_2^2 \leq 16 \text{ in} \\ & \text{Plain and Edge block,} \\ ec_{\text{intra}}^{\text{Wei}}(\mathbf{n}) & \\ \cdot \min \left(4, \max \left(\left(\frac{|C(\mathbf{k}, \mathbf{n})|}{t_{\text{lum}}^{\text{Wei}}(\mathbf{k}, \mathbf{n})} \right)^{0.36} \right) \right), & \text{otherwise,} \end{cases} \quad (4.37)$$

where $t_{\text{lum}}^{\text{Wei}}(\mathbf{k}, \mathbf{n})$ is the luminance adaptation adjusted threshold and $ec_{\text{intra}}^{\text{Wei}}(\mathbf{n})$ is the intra-band contrast masking. Equation (4.37) reveals that Wei's inter-band contrast masking for DCT coefficients at higher spatial frequencies ($k_1^2 + k_2^2 > 16$) is estimated at least four times higher than those estimated in Zhang's and Watson's PICs. Thus, significant blurring of some compressed images from Wei's PIC is likely to be caused by the over-estimation of the contrast masking at higher spatial frequencies.

The scatter plots for subjective rating versus perceptual distortion score of the four PICs are shown in Fig. 4.18, and their performance measure of the four PICs is tabulated in Table 4.9. The confidence interval of the Pearson correlation and Spearman rank-order correlation of the four PICs is shown in Fig. 4.19. The scatter plot in Fig. 4.18(a) clearly illustrates the limitation of Watson's PIC in correlating subjective rating with the widest spread of points among the four scatter plots. On this note, Watson's PIC has the largest confidence interval as shown in Fig. 4.19, and the lowest Pearson correlation and Spearman rank-order correlation in Table 4.9. On the other hand, the scatter plot of the proposed PIC illustrates the strongest positive correlation between the subjective rating and perceptual distortion score, and the proposed PIC also has the smallest confidence interval among the four PICs. Furthermore, the proposed PIC is found to have the highest Pearson correlation and Spearman rank-order correlation in Table 4.9.

Fig. 4.18 Scatter plots of subjective rating vs. subjective rating. Fitted polynomial function is shown in red. Scatter plots of Watson's, Wei's, Zhang's and the proposed PICs are shown in (a), (b), (c), and (d), respectively.

From Fig. 4.18, 4.19, and Table 4.9, it is found that Zhang's and Wei's PICs perform similarly, with Wei's PIC being marginally better than the Zhang's PIC. This observation coincides with those made on the noise shaping performance of the various JND models discussed in Section 4.2 (Table 4.3 and Fig. 4.11).

Based on the combination of perceptually significant subbands in each $N \times N$ sub-image, a DCT approximation is selected among ApD1 to ApD7. Therefore, the average computational cost of PSBDCT varies from image to image. To

Table 4.9 Performance measures of various PICs.

PIC	Performance Measure		
	ρ_p	ρ_s	r_e
Watson	0.619	0.629	0.171
Zhang	0.799	0.789	0.131
Wei	0.824	0.794	0.122
Proposed	0.858	0.849	0.105

Fig. 4.19 Pearson and Spearman rank-order correlations of the four PIC and its associated 95% confidence interval.

analyze the computational cost of PSBDCT, the average computational cost (ACC) of PSBDCT is defined as

$$\text{ACC} = \frac{N^2}{WH} \sum_{n_1=0}^{W/N-1} \sum_{n_2=0}^{H/N-1} \text{CC}(\mathbf{n}), \quad (4.38)$$

where $\text{CC}(\mathbf{n})$ is the computational cost of PSBDCT at the $\mathbf{n}$ th $N \times N$ sub-image.

Based on the computational cost analysis of the fast SBDCT algorithm in Tables 3.2 and 3.3, the computational cost of PSBDCT for TI-1 to TI-20 is computed and tabulated in Table 4.10, where FSA and FSA with $\mathbf{Q}$ are the computational cost of PSBDCT without and with quantization, respectively.

Based on the row-column decomposition method proposed by Loeffler *et al.* [LLM89], the computational cost of 8×8 DCT-II is (176, 464). Considering the computational cost of 8×8 DCT-II, PSBDCT requires fewer multiplications and additions as compared to Loeffler's fast DCT-II and the fast DCT-II algorithms discussed in Section 3.2.1.

Table 4.10 Average computational cost of PSBDCT.

Test-Image	Percentage of Dropped Subbands (%)				Average Computation Cost (c_m, c_a)	
	$c_{0,n}$	$c_{1,n}$	$c_{2,n}$	$c_{3,n}$	FSA	FSA with Q
TI-1	0.00	45.07	45.92	65.89	(128.83, 316.94)	(131.29, 316.94)
TI-2	0.00	43.02	48.34	75.49	(123.24, 125.62)	(125.62, 302.17)
TI-3	10.89	33.49	39.18	74.83	(124.55, 314.63)	(126.99, 314.63)
TI-4	0.00	12.79	38.40	65.45	(144.73, 368.26)	(148.06, 368.26)
TI-5	0.00	48.02	38.79	87.99	(118.27, 289.61)	(120.53, 289.61)
TI-6	0.00	50.15	67.29	91.14	(103.63, 105.55)	(105.55, 244.33)
TI-7	0.00	62.69	66.09	96.39	(95.93, 221.47)	(97.68, 221.47)
TI-8	0.34	48.54	65.19	96.24	(102.05, 241.04)	(103.98, 241.04)
TI-9	0.00	52.51	36.16	95.97	(112.96, 273.45)	(115.14, 273.45)
TI-10	7.76	34.39	34.28	57.74	(137.84, 349.39)	(140.50, 349.39)
TI-11	14.99	22.49	24.76	49.34	(146.91, 382.04)	(149.80, 382.04)
TI-12	0.00	41.41	37.69	70.29	(131.24, 326.36)	(133.91, 326.36)
TI-13	0.73	16.92	15.38	36.16	(169.67, 438.73)	(173.00, 438.73)
TI-14	0.00	30.93	21.78	86.43	(133.46, 336.94)	(136.08, 336.94)
TI-15	0.00	16.06	16.99	53.39	(160.17, 412.88)	(163.33, 412.88)
TI-16	0.00	44.58	30.66	73.09	(131.38, 326.11)	(133.93, 326.11)
TI-17	0.00	22.36	33.15	62.94	(145.41, 368.67)	(148.24, 368.67)
TI-18	0.00	13.18	9.13	52.76	(165.07, 428.10)	(168.32, 428.10)
TI-19	0.00	20.65	12.20	40.14	(167.33, 431.92)	(170.78, 431.92)
TI-20	0.00	63.57	57.69	95.36	(99.67, 234.56)	(101.50, 234.56)

4.4 Conclusions

This chapter presents a perceptually-tuned subband coder, which is based on the proposed FSA coupled with pixel- and DCT subband-based JND models. The pixel-based JND model is used to remove the perceptually insignificant subbands of each $N \times N$ sub-image. This method avoids the computation of the perceptually insignificant subbands without introducing perceptible distortion. The DCT

subband-based JND model together with a MTF is used to derive the base detection threshold. To adapt to the local activities of an image, the information from the pixel-based model is used to locally adapt the base detection threshold.

Many recent PICs [ZLX05, ZLX08, WN09] employ classification of each $N \times N$ sub-image (more commonly known as block classification) as plain, edge and texture regions of an image. This classification of each $N \times N$ sub-image facilitates better estimate of contrast masking, which had been validated by subjective evaluation in Section 4.2. Differing from Zhang's PIC, Wei's and the proposed PICs performed the block classification in the pixel domain, which was shown to be superior than the block classification performed in the DCT domain.

To perform block classification in the pixel domain, an edge map of an image is first computed. Taking advantage of the computational structure of the SBDCT, the proposed PIC directly estimates the edge map of an image using the subbands decomposed from each $N \times N$ sub-image, where the subbands are the intermediate result of SBDCT. Our approach avoids the complex computation required by the Canny edge operator adopted by Wei's PIC, and possesses comparable or better performance as compared to Zhang's and Wei's PICs. Using 20 test-images, the JND modeling of the proposed PIC is compared to another five PICs, and the proposed PIC was found to produce the least perceptible noise-contaminated images among the six PICs.

Error pooling is usually employed to obtain a single distortion score of a compressed image. For the proposed PIC, suitable values of β for spatial and frequency summation used in error pooling are determined by subjective evaluation. We found that the Zhang's error pooling strategy works well for the proposed PIC. To validate the compression performance of the proposed PIC, it is

compared with Watson's, Wei's, and Zhang's PICs using subjective evaluation of 20 test-images. These PICs are compared using the Pearson correlation and Spearman rank-order correlation. As discussed in Section 4.3, the proposed PIC produces the best compressed images at 0.4 and 0.5 bpp. Furthermore, the perceptual distortion score from the proposed PIC was found to be most correlated with subjective evaluation.

In the next chapter, we shall discuss the resizing of images and the integration of a JND model with image resizing.

Chapter 5

Perceptual Image Transcoding with SBDCT

With the emergence of mobile multimedia capable devices and availability of broadband networks, users have easy access to HD image and video contents. The mobile devices available to users usually have smaller displays as well as limited processing capabilities. Hence, an efficient method to reduce the spatial resolution of image or video content before delivery to these mobile devices is needed [VCS03], which results in shorter transmission time to deliver such contents. In addition, lower memory and processing requirements are imposed on these mobile devices.

Limited storage and bandwidth have long necessitated the use of compressed image and video contents, such as JPEG, MPEG-1/2/4, and H.261/3/4. It is highly desirable to be able to directly manipulate these contents in its compressed form. This approach eliminates the need to decompress and compress the image as well as video contents, which is necessary for pixel domain based techniques. Furthermore, it has been shown that resizing images in the DCT domain [DA01, PPO03, ST04, MM05, PP06, SC07, LL07] produces visually better images than pixel domain techniques.

Salazar and Tran [ST04] proposed an arbitrary transcoding algorithm in the DCT domain, which is a generalization of the method proposed by Dugad and Ahuja [DA01]. Their method combines up-sampling and IDCT-II, followed by down-sampling and forward DCT-II. Recently, Lee and Lin [LL07] proposed a method to downsize images by a factor of two with the flexibility of using

different number of DCT-II coefficients. They devised a computationally efficient scheme by using a sparse matrix representation similar to [DA01]. Park *et al.* [PPO03] used symmetric convolution to arbitrarily resize images in the DCT domain, where the multiplication-convolution property of DCT-II is exploited. Subsequently, Park *et al.* [PP06] proposed another transcoding method using fast inverse and forward DCT of composite length. Shu and Chau [SC07] proposed a two-stage structure to resize images in the DCT domain. Also, Shu and Chau determined the selection of parameters that specifies the amount of zero padding and truncation in their two-stage structure, which determines the amount of aliasing and the visual quality of the resized images. In [MM05], Mukherjee and Mitra proposed an image transcoding algorithm that employs the spatial relationship of DCT-II coefficients between a block and its sub-blocks [JF02].

To resize images by a factor of P/Q , where P and Q are integers, two schemes can be employed, which involves up-sampling by a factor of P followed by down-sampling by a factor of Q (up-down sampling), and vice versa (down-up sampling). In [MM05], the authors adopted the latter scheme to avoid high computational cost associated with the first scheme. However, resized images obtained using the latter scheme exhibit significant degradation [SC07]. In [PPO03, ST04, PP06, SC07], visually good images are obtained using the first scheme. Notably, the methods in [ST04, PP06, SC07] avoided high computational cost in the first scheme to resize images. However, the methods in [ST04, PP06, SC07] require mixed-radix DCTs, and may not yield efficient computation for all resizing factors. Salazar and Tran [ST04] proposed a computationally scalable approach, but they did not suggest the optimal length for DCT-II and IDCT-II for high PSNR.

Our work modifies the method that was proposed in [MM05]. Specifically, our method performs up-sampling by a factor of P and followed by down-sampling by a factor of Q to achieve resizing by a factor of P/Q . This scheme is avoided by Mukherjee and Mitra as it requires higher computational cost as compared to their method reported in [MM05]. On this note, we reduce the computational cost of our proposed method using a two-step approach. First, we generalize a symmetry property in the spatial relationship of DCT-II coefficients between a block and its sub-blocks for all resizing factors. Subsequently, a sparse matrix representation similar to [DA01] is derived from the generalized symmetry property. Second, further reduction of computational cost is achieved by using a subset of up-sampled DCT-II coefficients, instead of all up-sampled DCT-II coefficients, in the down-sampling operation. Our simulation results indicate that significant reduction of computational cost is achieved with little or no degradation in image quality as compared to the case where all up-sampled DCT-II coefficients are used in the down-sampling operation. This two-step approach leads to an efficient method to transcode images, and avoids DCT-II and IDCT-II in the down-sampling and up-sampling operations. In addition, the proposed method has a relatively regular structure, and does not require fast DCT-II and IDCT-II algorithms as compared to the methods reported in [ST04, PP06, SC07].

The rest of this chapter is structured as follows. In Section 5.1, the spatial relationship of the DCT-II coefficients between a block and its sub-blocks is reviewed. This is followed by the derivation of the generalized symmetry property in the relationship of the DCT-II coefficients between a block and its sub-blocks, which leads to the sparse matrix representation of the generalized symmetry property. Section 5.2 considers an image transcoding method to resize images by a

factor of P/Q . Using the sparse matrix representation and a subset of the up-sampled DCT-II coefficients, a faster image transcoding method is obtained. In order to analyze the distortion in the resized images, an equivalent filterbank of the image transcoding methods are derived and analyzed. Next, a perceptually-tuned image transcoding method is discussed in Section 5.3. Section 5.4 presents our simulation results and observations. Finally, our conclusions for this chapter are given in Section 5.5.

5.1 Spatial Relationship of DCT-II Coefficients between its Block and Sub-block

Using the spatial relationship of DCT-II coefficients between a block and its sub-blocks, an QN -point DCT-II block can be composed from Q neighboring N -point DCT-II blocks. Let $\mathbf{C}^{QN}$ denote a QN -point DCT-II block and $\mathbf{C}_{d_1}^{1D}$ denote an N -point DCT-II block, where $d_1 = 0, 1, \dots, Q-1$. The relationship for the 1-D case is

$$\mathbf{C}^{QN} = \mathbf{A}_{(Q,N)} \begin{bmatrix} \mathbf{C}_0^{1D} & \mathbf{C}_1^{1D} & \dots & \mathbf{C}_{Q-1}^{1D} \end{bmatrix}^t, \quad (5.1)$$

where $\mathbf{A}_{(Q,N)}$ is a $QN \times QN$ matrix. Let $a(j, k)$ denote the elements of $\mathbf{A}_{(Q,N)}$, where $j, k = 0, 1, \dots, QN-1$. The elements of $\mathbf{A}_{(Q,N)}$ are computed as

$$\begin{aligned} a(j, k) &= a(qN + u, pN + i) \\ &= \alpha^2 (qN + u) \sum_{v=0}^{N-1} W_c^{(2v+1)i/2N} W_c^{[2(pN+v)+1](qN+u)/2QN}, \end{aligned} \quad (5.2)$$

for $0 \leq p, q < Q, 0 \leq i, u < N$.

Let $\mathbf{C}^{PN \times QN}$ and $\mathbf{C}_{\mathbf{d}}^{2D}$ denote a $PN \times QN$ DCT-II block and an $N \times N$ DCT-II block, respectively, where $\mathbf{d} = \{d_1, d_2\}$, $d_1 = 0, 1, \dots, P-1$ and $d_2 = 0, 1, \dots, Q-1$.

The relationship for the 2-D case is stated as

$$\mathbf{C}^{PN \times QN} = \mathbf{A}_{(P,N)} \begin{bmatrix} \mathbf{C}_{0,0}^{2D} & \mathbf{C}_{0,1}^{2D} & \cdots & \mathbf{C}_{0,Q-1}^{2D} \\ \mathbf{C}_{1,0}^{2D} & \mathbf{C}_{1,1}^{2D} & \cdots & \mathbf{C}_{1,Q-1}^{2D} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}_{P-1,0}^{2D} & \mathbf{C}_{P-1,1}^{2D} & \cdots & \mathbf{C}_{P-1,Q-1}^{2D} \end{bmatrix} \mathbf{A}_{(Q,N)}^t. \quad (5.3)$$

Conversely, $P \times Q$ neighboring $N \times N$ DCT-II blocks can be decomposed from a $PN \times QN$ DCT-II block with the following equation:

$$\begin{bmatrix} \mathbf{C}_{0,0}^{2D} & \mathbf{C}_{0,1}^{2D} & \cdots & \mathbf{C}_{0,Q-1}^{2D} \\ \mathbf{C}_{1,0}^{2D} & \mathbf{C}_{1,1}^{2D} & \cdots & \mathbf{C}_{1,Q-1}^{2D} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}_{P-1,0}^{2D} & \mathbf{C}_{P-1,1}^{2D} & \cdots & \mathbf{C}_{P-1,Q-1}^{2D} \end{bmatrix} = \mathbf{A}_{(P,N)}^{-1} \mathbf{C}^{PN \times QN} \left(\mathbf{A}_{(Q,N)}^{-1} \right)^t. \quad (5.4)$$

Similar to [MM05], we shall refer the operations in (5.3) and (5.4) as composition and decomposition, respectively. Let $\mathbf{D}_{l_1, l_2}$ and $\mathbf{U}_{l_1, l_2}$ denote an $N \times N$ sub-matrix of $\mathbf{A}_{(Q,N)}$ and an $N \times N$ sub-matrix of $\mathbf{A}_{(Q,N)}^{-1}$, respectively, where $l_1, l_2 = 0, 1, \dots, Q-1$. Using these definitions of the $N \times N$ sub-matrices of $\mathbf{A}_{(Q,N)}$ and $\mathbf{A}_{(Q,N)}^{-1}$, the following equations are obtained:

$$\mathbf{A}_{(Q,N)} = \frac{1}{\sqrt{Q}} \begin{bmatrix} \mathbf{D}_{0,0} & \mathbf{D}_{0,1} & \cdots & \mathbf{D}_{0,Q-1} \\ \mathbf{D}_{1,0} & \mathbf{D}_{1,1} & \cdots & \mathbf{D}_{1,Q-1} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{D}_{Q-1,0} & \mathbf{D}_{Q-1,1} & \cdots & \mathbf{D}_{Q-1,Q-1} \end{bmatrix}, \quad (5.5)$$

and

$$\mathbf{A}_{(Q,N)}^{-1} = \sqrt{Q} \begin{bmatrix} \mathbf{U}_{0,0} & \mathbf{U}_{0,1} & \cdots & \mathbf{U}_{0,Q-1} \\ \mathbf{U}_{1,0} & \mathbf{U}_{1,1} & \cdots & \mathbf{U}_{1,Q-1} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{U}_{Q-1,0} & \mathbf{U}_{Q-1,1} & \cdots & \mathbf{U}_{Q-1,Q-1} \end{bmatrix}. \quad (5.6)$$

It is interesting to note that the sub-matrices of $\mathbf{A}_{(Q,N)}$ and $\mathbf{A}_{(Q,N)}^{-1}$ are always of $N \times N$ regardless of resizing factor Q , which simplifies the formulation of the proposed method presented in the following sections.

5.1.1 Generalization of Symmetry Property of Spatial Relationship of DCT-II Coefficients between its Block and Sub-blocks

In [JF02, MM05], sparseness of $\mathbf{A}_{(Q,N)}$ and $\mathbf{A}_{(Q,N)}^{-1}$ are exploited to achieve fast computation. Furthermore, Jiang and Feng [JF02] as well as Mukherjee and Mitra [MM05] observed that there are some elements of $\mathbf{A}_{(Q,N)}$ and $\mathbf{A}_{(Q,N)}^{-1}$ possess even and odd symmetries. Having these observations of $\mathbf{A}_{(Q,N)}$ and $\mathbf{A}_{(Q,N)}^{-1}$ in mind, we shall derive an efficient computational structure for $\mathbf{A}_{(Q,N)}$ and $\mathbf{A}_{(Q,N)}^{-1}$ in this section [TGW08, TGM11].

Generally, an image or a video frame is divided into $N \times N$ sub-images before any image or video processing takes place. Therefore, it is desired to formulate a general expression that describes the symmetry between $N \times N$ sub-matrices of $\mathbf{A}_{(Q,N)}$ for all values of Q and N . The symmetry of the elements in $\mathbf{A}_{(Q,N)}$ can be established by investigating the sub-matrices $\mathbf{D}_{l_1, l_2}$ and $\mathbf{D}_{l_1, Q-l_2-1}$, where $l_1, l_2 = 0, 1, \dots, Q/2 - 1$. For this task, we analyze the two cosine basis vectors in (5.2). The first cosine term in (5.2) is the basis vector of an N -point DCT-II. This basis vector possesses odd and even symmetry about $(N-1)/2$ when i is odd and even, respectively. Since i is the column index of $\mathbf{A}_{(Q,N)}$, the same basis vector is found in each column, and the basis vectors in the columns are repeated for every N columns.

The independent variables of the second cosine basis vector in (5.2) are p, q, u and v , where $p, q = 0, 1, \dots, Q-1$ and $u, v = 0, 1, \dots, N-1$. We consider $p = m$ and $p = Q - m - 1$, and the second basis vector $C_{2QN}^{[2(pN+v)+1](qN+u)}$ is re-expressed as

$$W_c^{[2(pN+v)+1](qN+u)/2QN} = \begin{cases} W_c^{(2mN+2v+1)j/2QN}, & 0 \leq p < \frac{Q}{2}, \\ (-1)^j W_c^{(2mN+2N-2v-1)j/2QN}, & \frac{Q}{2} \leq p < Q, \end{cases} \quad (5.7)$$

where $0 \leq j < QN$ and $j = qN + u$. We can further simplify (5.7) as

$$W_c^{[2(pN+v)+1](qN+u)/2QN} = \begin{cases} W_c^{(2mN+2v+1)j/2QN}, & 0 \leq p < \frac{Q}{2}, \\ (-1)^j W_c^{(2mN+2v'+1)j/2QN}, & \frac{Q}{2} \leq p < Q, \end{cases} \quad (5.8)$$

where $v' = N - 1 - v$. Equation (5.8) reveals $W_c^{[2(pN+v)+1](qN+u)/2QN}$ has even and odd symmetry about $QN/2$ for even and odd values of j , respectively.

The relation between $\mathbf{D}_{l_1, l_2}$ and $\mathbf{D}_{l_1, Q-l_2-1}$ is established by investigating the values of $W_c^{[2(pN+v)+1](qN+u)/2QN}$ when $p = m$ and $p = Q - m - 1$ for $v = 0, 1, \dots, N - 1$.

For even values of j , the values of $W_c^{[2(pN+v)+1](qN+u)/2QN}$ are identical when $p = m$

and $p = Q - m - 1$. For odd values of j , the values of $W_c^{[2(pN+v)+1](qN+u)/2QN}$ are also

identical except for a change of sign when $p = m$ and $p = Q - m - 1$. Together with

our earlier observations of the first basis vector in (5.2), we summarize our

findings as follows. First, the magnitude of the elements of $\mathbf{A}_{(Q, N)}$ is identical

when $p = m$ and $p = Q - m - 1$ for the same values of i and all values of j . Second,

the sign of the elements of $\mathbf{A}_{(Q, N)}$ is determined by the values of i and j , which is

summarized in Table 5.1. Based on these observations, the symmetry between the

sub-matrices of $\mathbf{A}_{(Q, N)}$ is determined and illustrated in Fig. 5.1. Using the same

approach, we found the same observations for sub-matrices $\mathbf{U}_{l_1, l_2}$ and $\mathbf{U}_{Q-l_1-1, l_2}$ of

$\mathbf{A}_{(Q, N)}^{-1}$.

Fig. 5.1 Relation between sub-matrices of $\mathbf{A}_{(Q,N)}$. Elements of $\mathbf{D}_{l_1, Q-l_2-1}$ can be obtained as the product of the corresponding entries in the relationship table and $\mathbf{D}_{l_1, l_2}$. The light and dark gray boxes in the relationship table represent 1 and -1 , respectively.

Table 5.1 Determining sign of elements of $\mathbf{A}_{(Q,N)}$.

i	j	N -point basis (Symmetry about $N/2$)	c_{2QN} (Symmetry about $Q/2$)	Elements at $p = m$ and $p = Q - m - 1$
Odd	Odd	Odd	Odd	Identical
Even	Odd	Even	Odd	Different Sign
Odd	Even	Odd	Even	Different Sign
Even	Even	Even	Even	Identical

By grouping the elements of $\mathbf{D}_{l_1, l_2}$ at the locations indicated by the light gray boxes in the relationship table shown in Fig. 5.1 into matrix $\mathbf{L}_{l_1, l_2}$, and the remaining elements into matrix $\mathbf{M}_{l_1, l_2}$, we have $\mathbf{D}_{l_1, l_2} = \mathbf{L}_{l_1, l_2} + \mathbf{M}_{l_1, l_2}$ and $\mathbf{D}_{l_1, Q-l_2-1} = \mathbf{L}_{l_1, l_2} - \mathbf{M}_{l_1, l_2}$. Likewise, by grouping the entries of $\mathbf{U}_{l_1, l_2}$ whose locations are indicated by the light gray boxes of the relationship table into matrix $\mathbf{V}_{l_1, l_2}$ and the remaining entries into matrix $\mathbf{W}_{l_1, l_2}$, we have $\mathbf{U}_{l_1, l_2} = \mathbf{V}_{l_1, l_2} + \mathbf{W}_{l_1, l_2}$ and $\mathbf{U}_{l_1, Q-l_2-1} = \mathbf{V}_{l_1, l_2} - \mathbf{W}_{l_1, l_2}$. Matrices $\mathbf{L}_{l_1, l_2}$, $\mathbf{M}_{l_1, l_2}$, $\mathbf{V}_{l_1, l_2}$, and $\mathbf{W}_{l_1, l_2}$ have only half nonzero elements as compared to $\mathbf{D}_{l_1, l_2}$ and $\mathbf{U}_{l_1, l_2}$, therefore $\mathbf{L}_{l_1, l_2}$, $\mathbf{M}_{l_1, l_2}$, $\mathbf{V}_{l_1, l_2}$, and $\mathbf{W}_{l_1, l_2}$ are referred as the sparse matrix representation of $\mathbf{D}_{l_1, l_2}$ and $\mathbf{U}_{l_1, l_2}$. Using the sparse matrix representation of $\mathbf{D}_{l_1, l_2}$ and $\mathbf{U}_{l_1, l_2}$, we rewrite (5.5) and (5.6) for odd values of Q as

$$\mathbf{A}_{(Q,N)} = \frac{1}{\sqrt{Q}} \begin{bmatrix} \mathbf{L}_{0,0} + \mathbf{M}_{0,0} & \cdots & \mathbf{L}_{0,\lfloor \frac{Q}{2} \rfloor - 1} + \mathbf{M}_{0,\lfloor \frac{Q}{2} \rfloor - 1} \\ \mathbf{L}_{1,0} + \mathbf{M}_{1,0} & \cdots & \mathbf{L}_{1,\lfloor \frac{Q}{2} \rfloor - 1} + \mathbf{M}_{1,\lfloor \frac{Q}{2} \rfloor - 1} \\ \vdots & \ddots & \vdots \\ \mathbf{L}_{Q-1,0} + \mathbf{M}_{Q-1,0} & \cdots & \mathbf{L}_{Q-1,\lfloor \frac{Q}{2} \rfloor - 1} + \mathbf{M}_{Q-1,\lfloor \frac{Q}{2} \rfloor - 1} \end{bmatrix} \quad (5.9)$$

$$\begin{bmatrix} \mathbf{L}_{0,\lfloor \frac{Q}{2} \rfloor} + \mathbf{M}_{0,\lfloor \frac{Q}{2} \rfloor} & \mathbf{L}_{0,\lfloor \frac{Q}{2} \rfloor - 1} - \mathbf{M}_{0,\lfloor \frac{Q}{2} \rfloor - 1} & \cdots & \mathbf{L}_{0,0} - \mathbf{M}_{0,0} \\ \mathbf{L}_{1,\lfloor \frac{Q}{2} \rfloor} + \mathbf{M}_{1,\lfloor \frac{Q}{2} \rfloor} & \mathbf{L}_{1,\lfloor \frac{Q}{2} \rfloor - 1} - \mathbf{M}_{1,\lfloor \frac{Q}{2} \rfloor - 1} & \cdots & \mathbf{L}_{1,0} - \mathbf{M}_{1,0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{L}_{Q-1,\lfloor \frac{Q}{2} \rfloor} + \mathbf{M}_{Q-1,\lfloor \frac{Q}{2} \rfloor} & \mathbf{L}_{Q-1,\lfloor \frac{Q}{2} \rfloor - 1} - \mathbf{M}_{Q-1,\lfloor \frac{Q}{2} \rfloor - 1} & \cdots & \mathbf{L}_{Q-1,0} - \mathbf{M}_{Q-1,0} \end{bmatrix},$$

and

$$\mathbf{A}_{(Q,N)}^{-1} = \sqrt{Q} \begin{bmatrix} \mathbf{V}_{0,0} + \mathbf{W}_{0,0} & \mathbf{V}_{0,1} + \mathbf{W}_{0,1} & \cdots & \mathbf{V}_{0,Q-1} + \mathbf{W}_{0,Q-1} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{V}_{\lfloor \frac{Q}{2} \rfloor - 1,0} + \mathbf{W}_{\lfloor \frac{Q}{2} \rfloor - 1,0} & \mathbf{V}_{\lfloor \frac{Q}{2} \rfloor - 1,1} + \mathbf{W}_{\lfloor \frac{Q}{2} \rfloor - 1,1} & \cdots & \mathbf{V}_{\lfloor \frac{Q}{2} \rfloor - 1,Q-1} + \mathbf{W}_{\lfloor \frac{Q}{2} \rfloor - 1,Q-1} \\ \mathbf{V}_{\lfloor \frac{Q}{2} \rfloor,0} + \mathbf{W}_{\lfloor \frac{Q}{2} \rfloor,0} & \mathbf{V}_{\lfloor \frac{Q}{2} \rfloor,1} + \mathbf{W}_{\lfloor \frac{Q}{2} \rfloor,1} & \cdots & \mathbf{V}_{\lfloor \frac{Q}{2} \rfloor,Q-1} + \mathbf{W}_{\lfloor \frac{Q}{2} \rfloor,Q-1} \\ \mathbf{V}_{\lfloor \frac{Q}{2} \rfloor - 1,0} - \mathbf{W}_{\lfloor \frac{Q}{2} \rfloor - 1,0} & \mathbf{V}_{\lfloor \frac{Q}{2} \rfloor - 1,1} - \mathbf{W}_{\lfloor \frac{Q}{2} \rfloor - 1,1} & \cdots & \mathbf{V}_{\lfloor \frac{Q}{2} \rfloor - 1,Q-1} - \mathbf{W}_{\lfloor \frac{Q}{2} \rfloor - 1,Q-1} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{V}_{0,0} - \mathbf{W}_{0,0} & \mathbf{V}_{0,1} - \mathbf{W}_{0,1} & \cdots & \mathbf{V}_{0,Q-1} - \mathbf{W}_{0,Q-1} \end{bmatrix}, \quad (5.10)$$

respectively. For even values of Q , (5.5) and (5.6) become

$$\mathbf{A}_{(Q,N)} = \frac{1}{\sqrt{Q}} \begin{bmatrix} \mathbf{L}_{0,0} + \mathbf{M}_{0,0} & \cdots & \mathbf{L}_{0,\lfloor \frac{Q}{2} \rfloor} + \mathbf{M}_{0,\lfloor \frac{Q}{2} \rfloor} & \mathbf{L}_{0,\lfloor \frac{Q}{2} \rfloor} - \mathbf{M}_{0,\lfloor \frac{Q}{2} \rfloor} \\ \mathbf{L}_{1,0} + \mathbf{M}_{1,0} & \cdots & \mathbf{L}_{1,\lfloor \frac{Q}{2} \rfloor} + \mathbf{M}_{1,\lfloor \frac{Q}{2} \rfloor} & \mathbf{L}_{1,\lfloor \frac{Q}{2} \rfloor} - \mathbf{M}_{1,\lfloor \frac{Q}{2} \rfloor} \\ \vdots & \ddots & \vdots & \vdots \\ \mathbf{L}_{Q-1,0} + \mathbf{M}_{Q-1,0} & \cdots & \mathbf{L}_{Q-1,\lfloor \frac{Q}{2} \rfloor} + \mathbf{M}_{Q-1,\lfloor \frac{Q}{2} \rfloor} & \mathbf{L}_{Q-1,\lfloor \frac{Q}{2} \rfloor} - \mathbf{M}_{Q-1,\lfloor \frac{Q}{2} \rfloor} \end{bmatrix} \quad (5.11)$$

$$\begin{bmatrix} \cdots & \mathbf{L}_{0,0} - \mathbf{M}_{0,0} \\ \cdots & \mathbf{L}_{1,0} - \mathbf{M}_{1,0} \\ \ddots & \vdots \\ \cdots & \mathbf{L}_{Q-1,0} - \mathbf{M}_{Q-1,0} \end{bmatrix},$$

and

$$\mathbf{A}_{(Q,N)}^{-1} = \sqrt{Q} \begin{bmatrix} \mathbf{V}_{0,0} + \mathbf{W}_{0,0} & \mathbf{V}_{0,1} + \mathbf{W}_{0,1} & \dots & \mathbf{V}_{0,Q-1} + \mathbf{W}_{0,Q-1} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{V}_{\lfloor \frac{Q}{2} \rfloor - 1,0} + \mathbf{W}_{\lfloor \frac{Q}{2} \rfloor - 1,0} & \mathbf{V}_{\lfloor \frac{Q}{2} \rfloor - 1,1} + \mathbf{W}_{\lfloor \frac{Q}{2} \rfloor - 1,1} & \dots & \mathbf{V}_{\lfloor \frac{Q}{2} \rfloor - 1,Q-1} + \mathbf{W}_{\lfloor \frac{Q}{2} \rfloor - 1,Q-1} \\ \mathbf{V}_{\lfloor \frac{Q}{2} \rfloor - 1,0} - \mathbf{W}_{\lfloor \frac{Q}{2} \rfloor - 1,0} & \mathbf{V}_{\lfloor \frac{Q}{2} \rfloor - 1,1} - \mathbf{W}_{\lfloor \frac{Q}{2} \rfloor - 1,1} & \dots & \mathbf{V}_{\lfloor \frac{Q}{2} \rfloor - 1,Q-1} - \mathbf{W}_{\lfloor \frac{Q}{2} \rfloor - 1,Q-1} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{V}_{0,0} - \mathbf{W}_{0,0} & \mathbf{V}_{0,1} - \mathbf{W}_{0,1} & \dots & \mathbf{V}_{0,Q-1} - \mathbf{W}_{0,Q-1} \end{bmatrix}, \quad (5.12)$$

respectively.

5.2 Image Transcoding Method to Resize Images

In [MM05], Mukherjee and Mitra proposed a method to resize images in the DCT domain, which applies subband approximation [JMM96]. Let $C^N(k_1)$ denote the coefficients of an N -point DCT-II block. Using subband approximation, an approximated PN -point DCT-II block $\hat{C}^{PN}(k_1)$ is

$$\hat{C}^{PN}(k_1) = \begin{cases} \sqrt{P} C^N(k_1), & 0 \leq k_1 \leq N-1, \\ 0, & N \leq k_1 < PN. \end{cases} \quad (5.13)$$

Similarly, an N -point DCT-II block can be approximated from a larger set of DCT-II coefficients by truncation. Thus, an approximated N -point DCT-II block $\hat{C}^N$ is computed as

$$\hat{C}^N(k_1) = C^{PN}(k_1) / \sqrt{P}, \quad 0 \leq k_1 \leq N-1. \quad (5.14)$$

5.2.1 Resizing Images by a Rational Factor of P/Q

In this section, we shall discuss about the proposed transcoding method based on (5.3) and (5.4) for a 1-D sequence, which can be easily extended to 2-D sequences.

Two transcoding methods to resize images by a factor of P/Q will be presented, where the second method is adapted from the first method to achieve more efficient computation with negligible loss of image quality. The main difference of the two methods is the size of the up-sampled DCT-II block used in the down-sampling stage, which greatly determines the computational cost of the method, and the visual quality of the resized images. Specifically, the first and second methods adopt block sizes denoted by N and $\tilde{N}$, respectively, where $N \geq \tilde{N}$.

Figure 5.2 shows the first transcoding method, adapted from [MM05], to resize a 1-D signal by a factor of P/Q , where P and Q are relatively prime integers larger than one. A total of Q consecutive N -point DCT-II blocks are required in this method. Each N -point DCT-II block is zero padded to a length of PN , and then decomposed into P N -point DCT-II blocks. These decomposed DCT-II blocks will be referred as the up-sampled DCT-II blocks for the rest of this chapter. Subsequently, Q up-sampled N -point DCT-II blocks are composed into an QN -point DCT-II block. Finally, each composed QN -point DCT-II block is truncated to a length of N , which is the resized DCT-II block.

To up-sample by a factor of P , an PN -point DCT-II block is first approximated from an N -point DCT-II block by zero padding. Subsequently, P consecutive N -point DCT-II blocks are decomposed from the approximated PN -point DCT-II block. Let $\hat{\mathbf{C}}_{d_1}^{PN}$ denote a zero padded PN -point DCT-II block obtained from $\mathbf{C}_{d_1}^{1D}$. Let $\mathbf{F}^{(d_1)}$ denote P consecutive N -point up-sampled DCT-II blocks, where $d_1 = 0, 1, \dots, Q-1$. From (5.6) and (5.13), we have

$$\mathbf{F}^{(d_1)} = \sqrt{P} \mathbf{A}_{(P,N)}^{-1} \hat{\mathbf{C}}_{d_1}^{PN}. \quad (5.15)$$

Fig. 5.2 Proposed method to resize a 1-D signal by a factor of P/Q using up-sampled DCT-II blocks each of size N . (a) Q N -point DCT-II blocks are used, (b) to (c) each DCT-II block is up-sampled by a factor of P , and (d) to (e) P consecutive Q up-sampled DCT-II blocks are down-sampled by a factor of Q .

Let $\mathbf{F}_{u_1}^{(d_1)}$ denote an N -point up-sampled DCT-II block in $\mathbf{F}^{(d_1)}$, where $u_1 = 0, 1, \dots,$

$P - 1$. From (5.6) and (5.13), the N -point up-sampled DCT-II blocks in $\mathbf{F}^{(d_1)}$ are obtained as

$$\mathbf{F}_{u_1}^{(d_1)} = P \mathbf{U}_{u_1,0} \mathbf{C}_{d_1}^{\text{1D}}. \quad (5.16)$$

To down-sample by a factor of Q , we first compose an QN -point DCT-II block from Q consecutive N -point DCT-II blocks. Subsequently, only the first N coefficients of the composed QN -point DCT-II block are retained. Let $\mathbf{G}$ denote a down-sampled N -point DCT-II block. Thus, down-sampling by a factor of Q is given as

$$\mathbf{G} = \frac{1}{\sqrt{Q}} \hat{\mathbf{A}}_{(Q,N)} \left[\mathbf{C}_0^{\text{1D}} \ \mathbf{C}_1^{\text{1D}} \ \dots \ \mathbf{C}_{Q-1}^{\text{1D}} \right]^t, \quad (5.17)$$

where $\hat{\mathbf{A}}_{(Q,N)}$ is

$$\hat{\mathbf{A}}_{(Q,N)} = \frac{1}{\sqrt{Q}} \begin{bmatrix} \mathbf{D}_{0,0} & \mathbf{D}_{0,1} & \cdots & \mathbf{D}_{0,Q-1} \end{bmatrix}. \quad (5.18)$$

Alternatively, (5.17) can be compactly written as

$$\mathbf{G} = \frac{1}{Q} \sum_{d_1=0}^{Q-1} \mathbf{D}_{0,d_1} \mathbf{C}_{d_1}^{\text{1D}}. \quad (5.19)$$

Let $\mathbf{G}_{u_1}$ denote an N -point DCT-II block obtained by resizing by a factor of P/Q .

Combining (5.16) and (5.19), resizing by a factor of P/Q is expressed by

$$\begin{aligned} \mathbf{G}_{u_1} &= \frac{1}{Q} \sum_{d_1=0}^{Q-1} \mathbf{D}_{0,d_1} \mathbf{F}_{\text{mod}_P(u_1 Q + d_1)}^{(\lfloor (u_1 Q + d_1)/P \rfloor)} \\ &= \frac{P}{Q} \sum_{d_1=0}^{Q-1} \mathbf{D}_{0,d_1} \mathbf{U}_{\text{mod}_P(u_1 Q + d_1),0} \mathbf{C}_{\lfloor (u_1 Q + d_1)/P \rfloor}^{\text{1D}}, \end{aligned} \quad (5.20)$$

where $\text{mod}_b(a)$ produces the remainder of a divided by b .

5.2.2 Reduction of Computational Cost by Sparse Matrix Representation

Using the sparse matrix representation of $\mathbf{D}_{0,d_1}$ and $\mathbf{D}_{0,Q-d_1-1}$, we can re-express

(5.19) as

$$\begin{aligned} \mathbf{G} &= \frac{1}{Q} \left[\sum_{d_1=0}^{\lfloor Q/2 \rfloor - 1} (\mathbf{D}_{0,d_1} \mathbf{C}_{d_1}^{\text{1D}} + \mathbf{D}_{0,Q-d_1-1} \mathbf{C}_{Q-d_1-1}^{\text{1D}}) + \mathbf{R}_{ds} \right] \\ &= \frac{1}{Q} \left\{ \sum_{d_1=0}^{\lfloor Q/2 \rfloor - 1} \left[\mathbf{L}_{0,d_1} (\mathbf{C}_{d_1}^{\text{1D}} + \mathbf{C}_{Q-d_1-1}^{\text{1D}}) + \mathbf{M}_{0,d_1} (\mathbf{C}_{d_1}^{\text{1D}} - \mathbf{C}_{Q-d_1-1}^{\text{1D}}) \right] + \mathbf{R}_{ds} \right\}, \end{aligned} \quad (5.21)$$

where

$$\mathbf{R}_{ds} = \begin{cases} 0, & Q \text{ is even,} \\ \mathbf{D}_{0,(Q-1)/2} \mathbf{C}_{(Q-1)/2}^{\text{1D}}, & Q \text{ is odd.} \end{cases} \quad (5.22)$$

Since there is only half the number of nonzero elements in $\mathbf{L}_{0,d_1}$ and $\mathbf{M}_{0,d_1}$ as compared to $\mathbf{D}_{0,d_1}$ and $\mathbf{D}_{0,Q-d_1-1}$, it is faster to compute (5.21) as compared to (5.19). Similarly, we can rewrite (5.16) as

$$\mathbf{F}_{u_1}^{(d_1)} = \begin{cases} P(\mathbf{V}_{u_1,0} + \mathbf{W}_{u_1,0})\mathbf{C}_{d_1}^{\text{1D}}, & 0 \leq u_1 < P/2, \\ P(\mathbf{V}_{P-u_1-1,0} - \mathbf{W}_{P-u_1-1,0})\mathbf{C}_{d_1}^{\text{1D}}, & P/2 \leq u_1 < P, \end{cases} \quad (5.23)$$

when P is even and

$$\mathbf{F}_{u_1}^{(d_1)} = \begin{cases} P(\mathbf{V}_{u_1,0} + \mathbf{W}_{u_1,0})\mathbf{C}_{d_1}^{\text{1D}}, & 0 \leq u_1 < \lfloor P/2 \rfloor, \\ P\mathbf{U}_{(P-1)/2,0}\mathbf{C}_{(P-1)/2}^{\text{1D}}, & u_1 = (P-1)/2, \\ P(\mathbf{V}_{P-u_1-1,0} - \mathbf{W}_{P-u_1-1,0})\mathbf{C}_{d_1}^{\text{1D}}, & \lfloor P/2 \rfloor < u_1 < P, \end{cases} \quad (5.24)$$

when P is odd. Instead of computing $\mathbf{F}_{u_1}^{(d_1)}$ using (5.16), a faster approach is to compute $\mathbf{F}_{u_1}^{(d_1)}$ using $\mathbf{V}_{u_1,0}\mathbf{C}_{d_1}^{\text{1D}}$ and $\mathbf{W}_{u_1,0}\mathbf{C}_{d_1}^{\text{1D}}$.

Table 5.2 compares the computational cost associated with up-sampling by factor of P and down-sampling by factor of Q between our proposed method and [MM05]. In this comparison, the same up-sampling and down-sampling factor are used for the horizontal and vertical directions. The computational cost in Table 5.2 is derived with $N = 8$. As observed from Table 5.2, the proposed method is also faster than [MM05] for all cases of P and Q . We achieve up to 57% reduction in computational cost with our method as compared to [MM05]. 1-D resizing by a factor of P/Q can be efficiently computed by combining using (5.20) to (5.23), and is expressed as

$$\begin{aligned}
\mathbf{G}_{u_1} &= \frac{P}{Q} \left\{ \sum_{d_1=0}^{\left\lfloor \frac{Q}{2} \right\rfloor - 1} \left[\mathbf{D}_{0,d_1} \mathbf{F}_{\text{mod}_P(u_1 Q + d_1)} + \mathbf{D}_{0,Q-1-d_1} \mathbf{F}_{\text{mod}_P((u_1+1)Q - d_1 - 1)} \right] + \mathbf{R} \right\} \\
&= \frac{P}{Q} \left\{ \sum_{d_1=0}^{\left\lfloor \frac{Q}{2} \right\rfloor - 1} \left[\mathbf{L}_{0,d_1} \left(\mathbf{F}_{\text{mod}_P(u_1 Q + d_1)} + \mathbf{F}_{\text{mod}_P((u_1+1)Q - d_1 - 1)} \right) \right. \right. \\
&\quad \left. \left. + \mathbf{M}_{0,d_1} \left(\mathbf{F}_{\text{mod}_P(u_1 Q + d_1)} - \mathbf{F}_{\text{mod}_P((u_1+1)Q - d_1 - 1)} \right) \right] + \mathbf{R} \right\}, \tag{5.25}
\end{aligned}$$

where

$$\mathbf{R} = \begin{cases} 0, & Q \text{ is even,} \\ \mathbf{D}_{0,(Q-1)/2} \mathbf{F}_{\text{mod}_P((u_1 Q + (Q-1)/2)/P)}, & Q \text{ is odd.} \end{cases} \tag{5.26}$$

Table 5.2 Computational cost of proposed method.

P	Computational cost per pixel (c_m, c_a) to down-sample by $1/P$		Computational cost per pixel (c_m, c_a) to up-sample by P	
	Proposed	Mukherjee [MM05]	Proposed	Mukherjee [MM05]
2	(3.00, 4.13)	(7.00, 6.00)	(3.00, 3.75)	(7.00, 5.25)
3	(3.33, 4.11)	(6.00, 5.50)	(3.33, 3.67)	(6.16, 4.72)
4	(3.75, 4.84)	(5.22, 7.38)	(3.75, 3.91)	(7.88, 6.56)
5	(3.60, 4.50)	(5.20, 6.42)	(3.60, 3.78)	(6.82, 5.58)
6	(3.31, 4.42)	(3.82, 6.60)	(3.31, 3.45)	(6.93, 5.74)
7	(3.43, 4.41)	(4.43, 6.39)	(3.43, 3.59)	(6.63, 5.51)
8	(3.94, 4.99)	(3.95, 7.78)	(3.94, 4.01)	(7.93, 6.89)

5.2.3 Reduction of Computational Cost by Using Subset of Up-Sampled DCT-II Coefficients

As mentioned earlier, resizing of images by a factor of P/Q is realized in two stages involving up-sampling and down-sampling by factor of P and Q , respectively. In this sub-section, we shall show that a faster computation scheme can be achieved by using a subset of up-sampled DCT-II coefficients in the down-sampling stage. This faster transcoding method is illustrated in Fig. 5.3.

Fig. 5.3 Faster proposed method to resize a 1-D signal by a factor of P/Q using a block size $\tilde{N}$ for each up-sampled DCT-II block. (a) Q N -point DCT-II blocks are used, (b) to (c) each DCT-II block is up-sampled by a factor of P , (d) the last $N - \tilde{N}$ up-sampled coefficients for each N -point DCT-II block are zeroed, (e) to (f) P sets of Q consecutive up-sampled DCT-II blocks are down-sampled by a factor of Q .

Using a subset of up-sampled DCT-II coefficients in the down-sampling stage is equivalent to reducing the number of nonzero entries in $\mathbf{D}_{0,d_1}$ and $\mathbf{U}_{u_1,0}$, as shown in Fig. 5.4. We denote these modified matrices in Fig. 5.4 as $\hat{\mathbf{D}}_{0,d_1}$ and $\hat{\mathbf{U}}_{u_1,0}$, respectively. Let $\hat{\mathbf{D}}_{0,d_1}$ denote an $N \times N$ matrix, where each row of $\hat{\mathbf{D}}_{0,d_1}$ has $\tilde{N}$ leftmost nonzero entries in the same row of $\mathbf{D}_{0,d_1}$. Let $\hat{\mathbf{U}}_{u_1,0}$ denote an $N \times N$ matrix, where each column of $\hat{\mathbf{U}}_{u_1,0}$ has $\tilde{N}$ topmost nonzero entries in the same column of $\mathbf{U}_{u_1,0}$. It is clear that the computational cost associated with $\mathbf{D}_{0,d_1}$ and

$\mathbf{U}_{u_1,0}$ are reduced by using $\hat{\mathbf{D}}_{0,d_1}$ and $\hat{\mathbf{U}}_{u_1,0}$, respectively. Also, the number of additions or subtractions in (5.25) involving $\mathbf{F}_{\text{mod}_P(u_1Q+d_1)}^{(\lfloor (u_1Q+d_1)/P \rfloor)} \pm \mathbf{F}_{\text{mod}_P((u_1+1)Q-d_1-1)}^{(\lfloor ((u_1+1)Q-d_1-1)/P \rfloor)}$ is reduced from N^2 to $N\tilde{N}$. A summary of the computational cost for the proposed method with block size $\tilde{N}$ is presented in Table 5.3, where $0 < \tilde{N} \leq 8$.

To analyze the distortion of the proposed method with block size $\tilde{N}$, we assumed that the proposed method with block size $\tilde{N}$ results in a different decimation filter as compared to the proposed method with block size N , but the interpolator filter remains unchanged. Considering (5.25) and (5.26), this assumption is valid since the resized DCT-II blocks obtained with $\hat{\mathbf{D}}_{0,d_1}$ and $\mathbf{U}_{u_1,0}$ are the same as those obtained with $\hat{\mathbf{D}}_{0,d_1}$ and $\hat{\mathbf{U}}_{u_1,0}$. Similar to [DA01], we represent the down-sampling stage using a filterbank structure shown in Fig. 5.5. The set of decimation filters in Fig. 5.5 is specified by a decimation filter matrix $\mathbf{H}$, which is determined by $\mathbf{D}_{0,d_1}$. By replacing $\hat{\mathbf{D}}_{0,d_1}$ with $\mathbf{D}_{0,d_1}$, another set of decimation filters is obtained and is represented by $\hat{\mathbf{H}}$.

In this sub-section, we report two studies of the degradation introduced by decimation filter matrix $\hat{\mathbf{H}}$. The first study investigates the response of the decimation filters given by $\hat{\mathbf{H}}$ with different values of $\tilde{N}$. The second study compares the resized images obtained by $\mathbf{H}$ and $\hat{\mathbf{H}}$. To analyze the decimation filters in the down-sampling stage, let $\mathbf{g}$ and $\mathbf{c}_i^{\text{1D}}$ denote the pixel domain equivalent of $\mathbf{G}$ and $\mathbf{C}_i^{\text{1D}}$, respectively. From (5.17), $\mathbf{g}$ is computed as

Fig. 5.4 A subset of up-sampled DCT-II coefficients considered in 1-D resizing. (a) Computational cost of the up-sampling stage is reduced using $\hat{U}_{u_1,0}$, where the bottom $(N \text{ to } \tilde{N}) \times N$ entries are zero and (b) computational cost of the down-sampling stage is reduced using $\hat{D}_{0,d_1}$, where the right $N \times (N - \tilde{N})$ entries are zero.

Table 5.3 Downsizing by factor of P/Q using various block size $\tilde{N}$.

$\tilde{N}$	Computational cost per pixel for P/Q (c_m, c_a)		
	$P/Q = 2/3$	$P/Q = 3/4$	$P/Q = 4/5$
1	(0.57, 1.43)	(1.18, 2.63)	(1.73, 3.93)
2	(1.56, 2.59)	(3.07, 5.15)	(4.50, 7.29)
3	(2.85, 4.25)	(5.60, 8.33)	(8.33, 11.86)
4	(4.58, 6.39)	(8.91, 12.09)	(13.20, 17.41)
5	(6.59, 8.68)	(12.79, 16.35)	(19.13, 23.95)
6	(9.06, 11.42)	(17.51, 21.35)	(26.10, 31.48)
7	(11.78, 14.37)	(22.75, 26.79)	(34.13, 39.99)
8	(15.00, 17.78)	(28.88, 33.03)	(43.20, 49.50)
$\tilde{N}$	Computational cost per pixel for P/Q (c_m, c_a)		
	$P/Q = 5/6$	$P/Q = 6/7$	$P/Q = 7/8$
1	(2.13, 5.27)	(2.70, 6.56)	(3.68, 7.98)
2	(5.56, 9.79)	(6.96, 11.91)	(9.35, 15.35)
3	(10.29, 15.50)	(12.76, 18.94)	(17.02, 24.45)
4	(16.32, 22.39)	(20.11, 27.43)	(26.69, 35.39)
5	(23.65, 30.47)	(29.01, 37.38)	(38.35, 48.19)
6	(32.29, 39.73)	(39.46, 48.78)	(52.01, 62.82)
7	(42.23, 50.17)	(51.45, 61.63)	(67.66, 79.29)
8	(53.47, 61.79)	(65.00, 75.94)	(85.31, 97.62)

Fig. 5.5. Filterbank structure of the down-sampling stage in the proposed method. N decimation filters are specified by the decimation filter matrix $\mathbf{H}$, h_m represents the m th row of $\mathbf{H}$, where $m = 0, 1, \dots, N-1$.

$$\begin{aligned}
 \mathbf{g} &= \mathbf{T}_N' \mathbf{G} \\
 &= \frac{1}{\sqrt{Q}} \mathbf{T}_N' \hat{\mathbf{A}}_{(Q,N)} [\mathbf{C}_0^{\text{ID}} \ \mathbf{C}_1^{\text{ID}} \ \dots \ \mathbf{C}_{Q-1}^{\text{ID}}]^t \\
 &= \frac{1}{\sqrt{Q}} \mathbf{T}_N' \hat{\mathbf{A}}_{(Q,N)} [\mathbf{T}_N \mathbf{c}_0^{\text{ID}} \ \mathbf{T}_N \mathbf{c}_1^{\text{ID}} \ \dots \ \mathbf{T}_N \mathbf{c}_{Q-1}^{\text{ID}}]^t \\
 &= \frac{1}{\sqrt{Q}} \mathbf{T}_N' \hat{\mathbf{A}}_{(Q,N)} (\mathbf{I}_Q \otimes \mathbf{T}_N) [\mathbf{c}_0^{\text{ID}} \ \mathbf{c}_1^{\text{ID}} \ \dots \ \mathbf{c}_{Q-1}^{\text{ID}}]^t,
 \end{aligned} \tag{5.27}$$

where $\mathbf{T}_N$ is the N -point DCT-II matrix, $\mathbf{I}_Q$ is the $Q \times Q$ identity matrix, and $\otimes$ denotes the Kronecker product. Hence, the decimation filter matrix $\mathbf{H}$ describing the down-sampling stage is given as

$$\begin{aligned}
 \mathbf{H} &= \frac{1}{\sqrt{Q}} \mathbf{T}_N' \hat{\mathbf{A}}_{(Q,N)} (\mathbf{I}_Q \otimes \mathbf{T}_N) \\
 &= \frac{1}{Q} [\mathbf{T}_N' \mathbf{D}_{0,0} \mathbf{T}_N \ \mathbf{T}_N' \mathbf{D}_{0,1} \mathbf{T}_N \ \dots \ \mathbf{T}_N' \mathbf{D}_{0,Q-1} \mathbf{T}_N].
 \end{aligned} \tag{5.28}$$

Similarly, we define $\hat{\mathbf{H}}$ as

$$\hat{\mathbf{H}} = \frac{1}{Q} [\mathbf{T}_N' \hat{\mathbf{D}}_{0,0} \mathbf{T}_N \ \mathbf{T}_N' \hat{\mathbf{D}}_{0,1} \mathbf{T}_N \ \dots \ \mathbf{T}_N' \hat{\mathbf{D}}_{0,Q-1} \mathbf{T}_N]. \tag{5.29}$$

Each decimation filter in Fig. 5.5 is a QN -tap filter, where h_m represents the m th row of $\mathbf{H}$. Let $C(z)$, $G(z)$, and $H_m(z)$ denote the z -transforms of $c(n)$, $g(n)$, and $h_m(n)$, respectively. Using standard results of multirate signal processing [Mit01], the output of the filterbank structure in Fig. 5.5 is described by

$$G(z) = \frac{1}{QN} \sum_{m=0}^{N-1} z^{-m} \sum_{k=0}^{QN-1} C(e^{-j2\pi k/QN} z^{1/Q}) H_m(e^{-j2\pi k/QN} z^{1/Q}), \quad (5.30)$$

and the equivalent filter of the filterbank structure is given by

$$F_k(z) = \sum_{m=0}^{N-1} z^{-Qm} H_m(z) e^{-j2\pi km/N}. \quad (5.31)$$

By replacing the decimation filter matrix $\mathbf{H}$ with $\hat{\mathbf{H}}$, we obtain the response of the modified decimation filters. For $N = 8$, Fig. 5.6(a) and 5.6(b) present the magnitude response of the equivalent filters for down-sampling factors of 3 and 4, respectively. The various filter responses in Fig. 5.6(a) and 5.6(b) are obtained by changing $\tilde{N}$ from 3 to 8. When N nonzero entries are used in each row of $\hat{\mathbf{D}}_{0,d_1}$, the response of $\mathbf{H}$ is obtained. By inspecting Fig. 5.6(a) and 5.6(b), it is found that the filter response of F_0 is similar when the number of nonzero entries in each row of $\hat{\mathbf{D}}_{0,d_1}$ ranges from 5 to 8. This observation suggests that the equivalent filter F_k can be realized with lesser nonzero entries in $\hat{\mathbf{D}}_{0,d_1}$ to reduce computational cost without affecting the visual quality of the resized images.

To effectively reduce computational cost of the proposed method, the smallest $\tilde{N}$ should be used with negligible distortion introduced in the resized images. Similar to the test methodology in [LL07], the downsized and resized images are obtained from the proposed method and its corresponding optimal least square upsizing filter [VS98], respectively, assuming P is smaller than Q . To measure the

Fig. 5.6 Magnitude response of equivalent filter defined by $\mathbf{H}$ ($N = 8$) and $\hat{\mathbf{H}}$ (a) for down-sampling by a factor of three, and (b) for down-sampling by a factor of four.

amount of distortion introduced by the down-sampling stage with various block size $\tilde{N}$, the same upsizing filter should be used for all values of $\tilde{N}$. To measure the distortion in downsized images, the downsized image is upsized by a factor of Q/P . The PSNR between the resized image (downsized then upsized) and the original

image is then computed [DA01, PPO03, ST04, MM05, PP06, SC07, LL07]. However, such an approach accounts for the distortion introduced in the downsizing and upsizing operations. To minimize the effects of the upsizing operation [VS98], an optimal least square upsizing filter is used to upsize the downsized images, and this upsizing filter is derived using $N = 8$ [TGM11].

Given a downsizing filter $\mathbf{J}$, the optimal least-square upsizing filter is found to be the Moore-Penrose pseudo-inverse of $\mathbf{J}$ [VS98], which is given as

$$\mathbf{J}^+ = \mathbf{J}^t (\mathbf{J}\mathbf{J}^t)^{-1}. \quad (5.32)$$

The downsizing filter $\mathbf{J}$ represents the equivalent filter to downsize an image by a factor of P/Q . By re-expressing (5.20) in its matrix representation, we have

$$\begin{bmatrix} \mathbf{G}_0 \\ \vdots \\ \mathbf{G}_{P-1} \end{bmatrix} = \frac{P}{Q} \left(\begin{bmatrix} \mathbf{D}_{0,0} & \cdots & \mathbf{D}_{0,Q-1} \end{bmatrix} \otimes \mathbf{I}_P \right) \left(\begin{bmatrix} \mathbf{U}_{0,0} \\ \vdots \\ \mathbf{U}_{P-1,0} \end{bmatrix} \otimes \mathbf{I}_Q \right) \begin{bmatrix} \mathbf{C}_0^{\text{1D}} \\ \vdots \\ \mathbf{C}_{Q-1}^{\text{1D}} \end{bmatrix}, \quad (5.33)$$

$\mathbf{J}$ is found to be

$$\mathbf{J} = \frac{P}{Q} \left(\begin{bmatrix} \mathbf{D}_{0,0} & \cdots & \mathbf{D}_{0,Q-1} \end{bmatrix} \otimes \mathbf{I}_P \right) \left(\begin{bmatrix} \mathbf{U}_{0,0} \\ \vdots \\ \mathbf{U}_{P-1,0} \end{bmatrix} \otimes \mathbf{I}_Q \right). \quad (5.34)$$

To analyze the effect of the block size $\tilde{N}$ in image resizing, PSNR and MSSIM values for 45 test-images (shown in Appendix B) are obtained. Table 5.4 and 5.5 present the PSNR and MSSIM values for downsizing factors of $2/3$ and $3/4$, respectively. As illustrated in Fig. 5.7, PSNR and MSSIM values are obtained from the original images and resized images that are downsized and subsequently upsized by factors of P/Q and Q/P , respectively. The PSNR and MSSIM values obtained with $N = 8$ are tabulated in the PSNR and MSSIM columns of Table 5.4 and 5.5, respectively. The PSNR and MSSIM values obtained with $\tilde{N} < 8$ are denoted as $\text{PSNR}(\tilde{N})$ and $\text{MSSIM}(\tilde{N})$, respectively.

Table 5.4 PSNR values for downsizing by factors of 2/3 and 3/4 with various block sizes.

Test Image	$P/Q = 2/3$				$P/Q = 3/4$			
	PSNR (dB)	PSNR – PSNR($\tilde{N}$) (dB)			PSNR (dB)	PSNR – PSNR($\tilde{N}$) (dB)		
		$\tilde{N} = 1$	$\tilde{N} = 2$	$\tilde{N} = 3$		$\tilde{N} = 1$	$\tilde{N} = 2$	$\tilde{N} = 3$
TI-1	24.45	6.61	3.20	0.26	26.21	6.78	1.37	0.03
TI-2	28.87	5.99	3.47	0.22	31.41	7.36	1.44	0.03
TI-3	32.38	8.36	3.53	0.29	33.94	7.90	1.32	0.03
TI-4	33.70	8.89	3.78	0.29	35.15	8.18	1.34	0.03
TI-5	35.93	11.25	5.59	0.32	37.05	10.94	1.99	0.02
TI-6	38.74	11.53	4.77	0.35	40.62	10.83	1.79	0.04
TI-7	38.19	11.69	5.32	0.28	40.44	11.70	1.85	0.02
TI-8	38.47	7.90	2.66	0.22	39.87	7.02	1.01	0.03
TI-9	36.20	9.25	4.05	0.27	38.00	8.92	1.36	0.03
TI-10	30.29	10.09	4.97	0.40	32.00	9.73	1.87	0.03
TI-11	31.01	8.70	3.64	0.28	32.78	8.44	1.46	0.03
TI-12	33.66	9.45	3.89	0.26	35.20	8.82	1.37	0.03
TI-13	25.73	5.67	2.37	0.25	27.01	5.49	1.06	0.03
TI-14	35.31	10.61	4.61	0.33	37.18	9.87	1.68	0.04
TI-15	28.56	6.53	2.84	0.26	29.96	6.28	1.13	0.03
TI-16	32.61	8.28	3.68	0.29	34.49	8.27	1.46	0.04
TI-17	30.04	8.00	3.65	0.29	31.73	7.94	1.34	0.03
TI-18	27.15	6.07	3.24	0.28	28.72	6.61	1.27	0.03
TI-19	34.08	5.82	1.65	0.15	35.06	4.84	0.71	0.02
TI-20	39.11	13.51	6.43	0.33	40.22	12.72	2.11	0.02
TI-21	48.09	14.61	5.81	0.39	49.67	13.22	2.03	0.04
TI-22	46.93	15.77	7.04	0.49	48.81	14.67	2.64	0.05
TI-23	44.71	16.40	7.29	0.52	47.04	15.62	3.07	0.06
TI-24	40.72	15.85	8.34	0.59	43.88	16.29	4.00	0.06
TI-25	38.77	15.72	8.57	0.65	41.90	16.28	4.15	0.06
TI-26	46.86	17.17	7.69	0.54	48.79	15.98	2.95	0.06
TI-27	45.19	14.73	8.10	0.58	47.93	14.99	3.53	0.06
TI-28	41.34	15.54	8.06	0.61	44.54	16.00	3.89	0.06
TI-29	46.13	15.12	6.96	0.43	48.28	14.39	2.63	0.05
TI-30	45.17	17.50	7.77	0.58	47.19	16.37	3.16	0.07
TI-31	42.33	18.20	9.29	0.67	45.07	17.96	4.34	0.08
TI-32	40.93	15.79	8.44	0.66	44.10	16.44	4.00	0.06
TI-33	29.81	8.04	4.31	0.32	32.26	8.68	1.80	0.03
TI-34	36.80	7.41	3.89	0.30	39.06	8.13	1.66	0.03
TI-35	38.50	9.18	4.59	0.33	40.72	9.55	1.91	0.04
TI-36	27.36	6.94	3.77	0.28	29.47	7.66	1.57	0.03
TI-37	31.17	10.02	5.15	0.36	34.02	10.89	2.36	0.04
TI-38	31.09	7.40	3.99	0.29	33.26	7.96	1.55	0.04
TI-39	38.36	11.89	5.38	0.36	40.75	11.85	2.34	0.04
TI-40	27.18	7.97	4.06	0.31	29.25	8.30	1.53	0.03
TI-41	34.38	8.14	4.13	0.31	36.70	8.89	1.71	0.03
TI-42	32.35	8.38	4.21	0.35	34.46	8.80	1.75	0.03
TI-43	32.80	7.86	4.03	0.30	35.07	8.42	1.67	0.03
TI-44	36.72	8.80	4.36	0.30	38.49	8.40	1.60	0.04
TI-45	33.42	9.12	4.69	0.34	35.70	9.62	1.94	0.03
Mean	35.81	10.62	5.05	0.37	37.85	10.53	2.06	0.04

Table 5.5 MSSIM values for downsizing by factors of 2/3 and 3/4 with various block sizes.

Test Image	$P/Q = 2/3$				$P/Q = 3/4$			
	MSSIM	MSSIM – MSSIM($\tilde{N}$)			MSSIM	MSSIM – MSSIM($\tilde{N}$)		
		$\tilde{N} = 1$	$\tilde{N} = 2$	$\tilde{N} = 3$		$\tilde{N} = 1$	$\tilde{N} = 2$	$\tilde{N} = 3$
TI-1	0.880	0.30	0.08	0.00	0.912	0.21	0.00	0.00
TI-2	0.930	0.31	0.11	0.00	0.958	0.24	0.01	0.00
TI-3	0.922	0.30	0.07	0.00	0.943	0.20	0.02	0.00
TI-4	0.920	0.26	0.06	0.00	0.938	0.17	0.01	0.00
TI-5	0.974	0.26	0.06	0.00	0.981	0.16	0.00	0.00
TI-6	0.963	0.19	0.04	0.00	0.973	0.12	0.01	0.00
TI-7	0.977	0.17	0.03	0.00	0.984	0.10	0.00	0.00
TI-8	0.936	0.16	0.04	0.00	0.951	0.11	0.01	0.00
TI-9	0.959	0.28	0.06	0.00	0.973	0.18	0.01	0.00
TI-10	0.940	0.41	0.10	0.00	0.958	0.26	0.02	0.00
TI-11	0.915	0.35	0.10	0.01	0.943	0.24	0.02	0.00
TI-12	0.911	0.24	0.06	0.00	0.932	0.17	0.01	0.00
TI-13	0.842	0.34	0.12	0.01	0.889	0.27	0.03	0.00
TI-14	0.941	0.30	0.07	0.00	0.958	0.19	0.01	0.00
TI-15	0.889	0.35	0.10	0.01	0.919	0.25	0.02	0.00
TI-16	0.937	0.24	0.06	0.00	0.957	0.16	0.01	0.00
TI-17	0.918	0.33	0.09	0.00	0.941	0.23	0.02	0.00
TI-18	0.905	0.44	0.14	0.01	0.936	0.31	0.02	0.00
TI-19	0.853	0.19	0.06	0.01	0.885	0.16	0.02	0.00
TI-20	0.985	0.18	0.03	0.00	0.988	0.11	0.00	0.00
TI-21	0.990	0.08	0.01	0.00	0.992	0.05	0.00	0.00
TI-22	0.990	0.10	0.01	0.00	0.993	0.06	0.00	0.00
TI-23	0.987	0.13	0.02	0.00	0.991	0.07	0.00	0.00
TI-24	0.987	0.25	0.04	0.00	0.993	0.14	0.00	0.00
TI-25	0.988	0.29	0.05	0.00	0.993	0.16	0.01	0.00
TI-26	0.990	0.14	0.02	0.00	0.993	0.07	0.00	0.00
TI-27	0.990	0.19	0.04	0.00	0.994	0.11	0.00	0.00
TI-28	0.987	0.25	0.04	0.00	0.993	0.14	0.00	0.00
TI-29	0.989	0.11	0.02	0.00	0.992	0.06	0.00	0.00
TI-30	0.991	0.20	0.03	0.00	0.994	0.11	0.00	0.00
TI-31	0.991	0.31	0.05	0.00	0.995	0.17	0.01	0.00
TI-32	0.987	0.25	0.04	0.00	0.993	0.14	0.00	0.00
TI-33	0.908	0.43	0.14	0.01	0.945	0.30	0.02	0.00
TI-34	0.944	0.20	0.07	0.00	0.966	0.15	0.01	0.00
TI-35	0.969	0.16	0.05	0.00	0.980	0.11	0.01	0.00
TI-36	0.896	0.46	0.16	0.01	0.938	0.34	0.03	0.00
TI-37	0.952	0.40	0.10	0.00	0.974	0.26	0.01	0.00
TI-38	0.923	0.33	0.11	0.01	0.954	0.24	0.02	0.00
TI-39	0.979	0.20	0.03	0.00	0.987	0.11	0.00	0.00
TI-40	0.907	0.40	0.12	0.00	0.943	0.29	0.02	0.00
TI-41	0.945	0.28	0.08	0.00	0.967	0.20	0.01	0.00
TI-42	0.948	0.25	0.07	0.00	0.967	0.18	0.01	0.00
TI-43	0.933	0.28	0.09	0.00	0.958	0.21	0.01	0.00
TI-44	0.953	0.19	0.06	0.00	0.970	0.13	0.01	0.00
TI-45	0.944	0.34	0.10	0.00	0.966	0.23	0.01	0.00
Mean	0.946	0.263	0.067	0.002	0.963	0.175	0.009	0.000

Fig. 5.7 Downsizing and upscaling of images for image comparison using PSNR and MSSIM.

From Table 5.4, it is observed that $\tilde{N} > 3$ and $\tilde{N} > 2$ yield little or no improvement in PSNR values for downsizing by factors of 2/3 and 3/4, respectively. Similar observations are found with the MSSIM values in Table 5.5. It is observed that $\tilde{N}$ more than two and one yields little or no improvement in MSSIM values for downsizing by a factor of 2/3 and 3/4, respectively.

To further investigate on the degradation of the downsized images with respect to $\tilde{N}$, we examine the differential PSNR (DPSNR) and the differential MSSIM (DMSSIM) values for other downsizing factors. Since the maximum achievable PSNR and MSSIM values for any downsizing factor are obtained when $\hat{\mathbf{H}} = \mathbf{H}$ ($\tilde{N} = 8$), we define the average DPSNR $\text{ADP}(\tilde{N})$ and the average DMSSIM $\text{ADM}(\tilde{N})$ as

$$\text{ADP}(\tilde{N}) = \frac{\text{PSNR} - \text{PSNR}(\tilde{N})}{N_{\text{TI}}}, \quad (5.35)$$

and

$$\text{ADM}(\tilde{N}) = \frac{\text{MSSIM} - \text{MSSIM}(\tilde{N})}{N_{\text{TI}}}, \quad (5.36)$$

where N_{TI} is the number of test-images.

Figure 5.8 presents the plots of $\text{ADP}(\tilde{N})$ and $\text{ADM}(\tilde{N})$ versus block size $\tilde{N}$ for images down-sampled by a factor of Q , where $Q = 1, 2, \dots, 13$. It is interesting

Fig. 5.8 Average differential PSNR and MSSIM of test images down-sampled by a factor of Q . (a) Average differential PSNR for $Q = 2, 3, \dots, 7$ and (b) average differential PSNR for $Q = 8, 9, \dots, 13$. (c) Average differential MSSIM for $Q = 2, 3, \dots, 7$ and (d) average differential MSSIM for $Q = 8, 9, \dots, 13$.

to note that the plots of $\text{ADP}(\tilde{N})$ and $\text{ADM}(\tilde{N})$ show similar trends, therefore we determine the selection of $\tilde{N}$ using $\text{ADP}(\tilde{N})$ only. The plots of $\text{ADM}(\tilde{N})$ for $2 \leq P \leq 11$, $3 \leq Q \leq 12$, are presented in Appendix C. In this section, we will determine the value of $\tilde{N}$ with respect to values of P and Q , which produces negligible or no difference in terms of PSNR values (≤ 0.05 dB) as compared to the block size N .

First, we investigate the changes of $\text{ADP}(\tilde{N})$ with respect to Q and $\tilde{N}$. From Fig. 5.8(a) and 5.8(b), it is clear that reduction of $\text{ADP}(\tilde{N})$ diminishes when Q

and $\tilde{N}$ increase. Hence, the highest value of $\text{ADP}(\tilde{N})$ is observed when $Q = 2$ for all values of $\tilde{N}$. Next, we investigate the changes of $\text{ADP}(\tilde{N})$ with respect to P and $\tilde{N}$. The plots in Fig. 5.9 and 5.10 present $\text{ADP}(\tilde{N})$ for a fixed down-sampling factor Q and a variable up-sampling factor P . From the plots in Fig. 5.9 and 5.10, it is observed that there is little or negligible reduction of $\text{ADP}(\tilde{N})$ for resizing factor P/Q when $\tilde{N}$ is increased beyond three and $1 \leq P \leq 4$, $3 \leq Q \leq 5$. Also, there is little or negligible reduction of $\text{ADP}(\tilde{N})$ for resizing factor P/Q when $\tilde{N}$ is increased beyond two and $1 \leq P \leq 11$, $6 \leq Q \leq 12$. From Fig. 5.9 and 5.10, it is clear that increasing $\tilde{N}$ yields smaller values of $\text{ADP}(\tilde{N})$ for all values of P , and $\text{ADP}(\tilde{N})$ increases as P increases. Thus, there is a direct relation between P and $\text{ADP}(\tilde{N})$, and an inverse relation between Q and $\text{ADP}(\tilde{N})$. From these observations made from Fig. 5.8 to 5.10, it becomes apparent that the lowest value of $\tilde{N}$ to achieve $\text{ADP}(\tilde{N}) < 0.05$ is obtained when $P = 1$ for all values of Q .

To obtain the lowest computational cost, the lowest value of $\tilde{N}$ to achieve $\text{ADP}(\tilde{N}) < 0.05$ is to be determined, and this lowest value of $\tilde{N}$ to produce $\text{ADP}(\tilde{N}) < 0.05$ is denoted as $N_{0.05}$. Using the observations made from Fig. 5.8 to 5.10, we empirically determined the values of $N_{0.05}$ to achieve $\text{ADP}(N_{0.05}) < 0.05$ for resizing factor P/Q , and $N_{0.05}$ is expressed as

Fig. 5.9 Average PSNR difference for various downsizing factor P/Q for $Q = 3$ to 7 in (a) to (e), respectively.

Fig. 5.10 Average PSNR difference for various downsizing factor P/Q for $Q = 8$ to 12 in (a) to (e), respectively.

$$N_{0.05} = \begin{cases} 5, & Q = 2, \\ 4, & Q = 3, \\ 3, & 4 \leq Q \leq 5, \\ 3, & Q = 6, P > 1, \\ 2, & Q = 6, P = 1, \\ 2, & Q \geq 7, \end{cases} \quad (5.37)$$

where $1 \leq P \leq 11$, $3 \leq Q \leq 12$.

5.3 Perceptual Image Transcoding

This section presents an alternative approach to determine the value of $\tilde{N}$ using the base visibility threshold derived in Chapter 4. For this approach, we select the perceptually significant DCT-II coefficients in each $N \times N$ DCT-II block to be used in the down-sampling stage, instead of a fixed number of DCT-II coefficients (given by $N_{0.05}$) for each $N \times N$ DCT-II block. Since the content of each $N \times N$ DCT-II block can be different, different values of $\tilde{N}$ for each $N \times N$ DCT-II block should be used to further reduce computational cost. Therefore, a perceptually-tuned approach can effectively reduce the number of up-sampled DCT-II coefficients used in the down-sampling operation without degrading the resized images. On this note, this perceptually-tuned approach is different from the one discussed in Section 5.2.3 as the selection of the block size $\tilde{N}$ takes into account the spatial frequency sensitivity of a human viewer, as well as the content of the up-sampled DCT-II coefficients.

As illustrated in Fig. 5.2 and 5.3, the proposed image transcoding method first generates the up-sampled DCT-II coefficients from the original DCT-II coefficients, and subsequently derives the down-sampled DCT-II coefficients from the up-sampled DCT-II coefficients. In this section, we shall investigate on

the feasibility of using the base visibility threshold to select the perceptually significant up-sampled DCT-II coefficients to be applied in the down-sampling stage. The objective of such a perceptually-tuned transcoding method is to determine the least number of up-sampled DCT-II coefficients in perceptual sense, so that visually good resized images can be achieved at the lowest computational cost.

As discussed in Chapter 4, the viewing distance of the processed image is defined as six times of the image height, and the base visibility threshold $T_{b,MTF}(\mathbf{k})$ is defined in (4.17). As mentioned earlier in the introduction of this chapter, the proposed image transcoding method does not require fast DCT-II and IDCT-II algorithms. This is due to the fact that the up-sampled and down-sampled DCT-II blocks have the same basis vector as the original DCT-II block regardless of the resizing factor P/Q . However, the viewing distance of an up-sampled image is dependent on the up-sampling factor P , therefore, different $N \times N$ base visibility threshold $T_{b,MTF}(\mathbf{k})$ is required for different up-sampling factor P .

To reduce the number of up-sampled DCT-II coefficients used in the down-sampling stage, we use the threshold $T_{b,MTF}(\mathbf{k})$ to determine the perceptually significant up-sampled DCT-II coefficients (coefficients higher than threshold). The resultant block size used in the perceptually-tuned transcoding method is denoted as N_{PT} . Contrary to $N_{0.05}$, N_{PT} varies from one $N \times N$ up-sampled DCT-II block to another, depending on the number of perceptually significant DCT-II coefficients in each $N \times N$ up-sampled DCT-II block.

A subjective evaluation is conducted with 10 subjects to compare the test-images TI-1 to TI-20 downsized by the perceptually-tuned transcoding method

and those downsized by the transcoding method discussed in Section 5.2.1. Specifically, each downsized image obtained using the perceptually-tuned transcoding method is compared with the corresponding downsized image obtained using the proposed transcoding method with block size N , where $N = 8$. From the subjective evaluation, it is found that the downsized images from these two methods are visually identical at a viewing distance of six times of the image height, for values of $P = 2, 3, \dots, 11$.

To obtain the perceptual distortion score P_{PT} of a downsized image obtained using the perceptually-tuned transcoding method, the same methodology discussed in Section 5.2.3 is used. The perceptual distortion score and PSNR are computed using the original test-images, and the resized images are obtained using the image transcoding method and its optimal least square upsizing filter as shown in Fig. 5.7. In this section, the resized images obtained using the perceptually-tuned transcoding method with block size N_{PT} are compared with the resized images obtained using the proposed transcoding method with block size N and $N_{0.05}$. For the rest of this chapter, the proposed transcoding method with block sizes N , $N_{0.05}$ and N_{PT} are referred as $\text{prop}(N)$, $\text{prop}(N_{0.05})$, and $\text{prop}(N_{PT})$, respectively. The perceptual distortion scores of $\text{prop}(N_{0.05})$ and $\text{prop}(N)$ are referred as $P_{0.05}$ and P_N , respectively.

Tables 5.6 and 5.7 summarize the perceptual distortion score and PSNR for downsizing test-images (TI-1 to TI-20) using $\text{prop}(N)$, $\text{prop}(N_{0.05})$, and $\text{prop}(N_{PT})$, respectively, where $P/Q = 2/3$ and $3/4$. Most of the perceptual distortion score and PSNR of $\text{prop}(N_{PT})$ are found to similar to $\text{prop}(N)$. Small

Table 5.6 PSNR and perceptual distortion score of images downsized by a factor of 2/3.

Test Image	prop(N)		prop($N_{0.05}$)		prop(N_{PT})	
	PSNR (dB)	P_N	PSNR (dB)	$P_{0.05}$	PSNR (dB)	P_{PT}
TI-1	24.45	83.16	24.44	83.52	24.43	83.28
TI-2	28.87	36.89	28.86	37.29	28.84	37.36
TI-3	32.38	76.68	32.38	76.16	32.26	75.95
TI-4	33.70	35.97	33.70	36.05	33.56	35.75
TI-5	35.93	34.84	35.92	35.00	35.83	34.71
TI-6	38.74	21.38	38.74	21.53	38.36	22.44
TI-7	38.19	31.18	38.18	31.17	38.04	31.31
TI-8	38.47	21.91	38.47	22.21	38.10	22.37
TI-9	36.20	36.49	36.20	36.80	36.10	36.60
TI-10	30.29	44.57	30.28	44.38	30.21	44.41
TI-11	31.01	52.30	31.01	52.34	30.92	51.79
TI-12	33.66	24.61	33.65	24.69	33.50	24.63
TI-13	25.73	45.11	25.72	44.98	25.69	44.76
TI-14	35.31	29.10	35.31	29.63	35.09	29.33
TI-15	28.56	38.11	28.55	38.19	28.50	38.16
TI-16	32.61	36.08	32.60	36.31	32.51	36.59
TI-17	30.04	49.64	30.04	49.77	29.97	49.84
TI-18	27.15	48.87	27.14	48.91	27.11	48.59
TI-19	34.08	19.76	34.07	20.07	33.90	20.24
TI-20	39.11	30.14	39.11	30.08	38.97	30.32
Mean	32.72	39.84	32.72	39.95	32.59	39.92

differences between $P_{0.05}$, P_{PT} and P_N , suggest that the downsized images obtained using the proposed transcoding method having block sizes $N_{0.05}$, N_{PT} , and N are visually similar. This observation agrees well with our subjective evaluation of the downsized images from $\text{prop}(N)$, $\text{prop}(N_{0.05})$, and $\text{prop}(N_{PT})$. The average PSNR and perceptual distortion score of the resized images obtained with $\text{prop}(N)$, $\text{prop}(N_{0.05})$, and $\text{prop}(N_{PT})$, where $P/Q = 2/3, 3/4, 4/5, 5/6, 6/7$, and $7/8$ are shown in Table 5.8, and the 95% confidence interval of the perceptual distortion score of the resized images is shown in Fig. 5.11. From Table 5.8, it is noted that there is an increase in the differences

Table 5.7 PSNR and perceptual distortion score of images downsized by a factor of 3/4.

Test Image	prop(N)		prop($N_{0.05}$)		prop(N_{PT})	
	PSNR (dB)	P_N	PSNR (dB)	$P_{0.05}$	PSNR (dB)	P_{PT}
TI-1	26.21	71.22	26.18	70.38	26.16	70.45
TI-2	31.41	28.24	31.39	28.45	31.31	28.55
TI-3	33.94	55.97	33.91	55.71	33.71	53.67
TI-4	35.15	31.06	35.12	30.88	34.90	30.43
TI-5	37.05	28.33	37.03	28.42	36.86	28.72
TI-6	40.62	18.65	40.59	18.57	39.97	18.02
TI-7	40.44	22.51	40.42	22.79	40.12	22.38
TI-8	39.87	18.83	39.84	18.75	39.27	18.51
TI-9	38.00	28.97	37.97	29.12	37.76	29.20
TI-10	32.00	42.97	31.97	42.60	31.83	41.79
TI-11	32.78	53.26	32.75	52.25	32.58	50.36
TI-12	35.20	19.78	35.17	19.79	34.90	18.99
TI-13	27.01	42.84	26.98	42.85	26.93	42.43
TI-14	37.18	23.29	37.15	23.11	36.76	22.85
TI-15	29.96	32.76	29.93	32.56	29.83	32.33
TI-16	34.49	28.14	34.46	27.99	34.28	28.15
TI-17	31.73	42.26	31.70	41.84	31.58	41.91
TI-18	28.72	41.18	28.69	41.11	28.64	41.09
TI-19	35.06	15.80	35.03	15.95	34.75	15.04
TI-20	40.22	23.55	40.20	23.63	39.94	24.17
Mean	34.35	33.48	34.32	33.34	34.10	32.95

Table 5.8 Mean PSNR and perceptual distortion score for downsizing of images by a factor of P/Q .

P/Q	prop(N)		prop($N_{0.05}$)		prop(N_{PT})	
	PSNR (dB)	P_N	PSNR (dB)	$P_{0.05}$	PSNR (dB)	P_{PT}
2/3	32.72	39.84	32.72	39.95	32.59	39.92
3/4	34.35	33.48	34.32	33.34	34.10	32.95
4/5	35.38	34.72	35.38	34.61	34.98	33.31
5/6	36.29	32.51	36.29	32.51	35.82	31.37
6/7	37.11	32.61	37.06	32.35	36.56	32.17
7/8	38.07	25.74	38.03	25.59	37.31	25.26

between the PSNR values of prop(N) and prop(N_{PT}) as P increases. However, the differences between perceptual distortion score $P_{0.05}$, P_{PT} , and P_N remain small regardless of the value of P .

Fig. 5.11 Perceptual distortion score and its associated 95% confidence interval of images resized by a factor of P/Q .

The base visibility threshold is dependent on the viewing distance (product of the image height and P), therefore, the base visibility threshold increases when P increases, which in turn leads to more aggressive thresholding of up-sampled DCT-II coefficients. Hence, it is expected that there would be increasing difference between the PSNR values of $\text{prop}(N)$ and $\text{prop}(N_{PT})$ as P increases. However, the perceptual distortion score of the three transcoding methods remains similar. In fact, the confidence interval of the perceptual distortion score of the three transcoding methods indicates that slightly lower perceptual distortion score is obtained with $\text{prop}(N_{PT})$ for resizing factor 3/4, 4/5, 5/6, and 6/7. Based on our observations from Fig. 4.15(b), a marginal difference of perceptual distortion score (less than ten) will not lead to perceptible difference. This conclusion is reinforced by our subjective evaluation, which revealed no visible difference between the resized images from the three transcoding methods.

5.4 Results and Discussion

In this section, we present a few performance comparisons between the proposed, Park's [PP06], and Shu's [SC07] transcoding methods for downsizing images by a factor of P/Q . A generalization of the transcoding methods proposed in [PP06] and [SC07] is given in Fig. 5.12. In [PP06], Park *et al.* proposed two transcoding methods, which impose several constraints specifying the amount of zero padding and truncation required in the up-sampling and down-sampling stages of their methods. These two transcoding methods are referred as park(I) and park(II), where park(II) is faster to compute as compared to park(I) at the expense of visual quality of the downsized images. In park(I), Park *et al.* introduced two constraints, which are given as $\tilde{N} = \text{common multiple of } \{P, Q\}$ and $\tilde{N} \geq \text{Max}\{8P, 8Q\}$, where $\tilde{N} = Q(8+q)$ and $q \geq 0$. The value of q specifies the amount of zero padding required by each 8-point DCT-II block to a length of P' , where $P' = 8 + q$. By replacing $\tilde{N} \geq \text{Max}\{8P, 8Q\}$ with $\text{Max}\{8P, 8Q\} > \tilde{N} \geq \text{Min}\{8P, 8Q\}$, Park *et al.* reduced the computational cost of park(II) for some resizing factors. Using these constraints, the length of the DCT-II and IDCT-II required by Park's methods is determined as $\tilde{N}/Q$ and P' , respectively.

Shu and Chau [SC07] proposed several combinations of zero padding and truncation in the up-sampling and down-sampling stages of their methods to produce downsized images having different visual quality. They suggested various block sizes that differs in computational cost and image quality. In this section, we adopt $P' = 8P$, $Q' = 8Q$, and $P' = 4P$, $Q' = 4Q$, as Case I and Case II, respectively, as they are found to produce visually good images at low computational cost [SC07]. These two cases of the Shu's method are referred as

Fig. 5.12 Generalized operations of the methods proposed by Park [PP06] and Shu [SC07].

shu(I) and shu(II), respectively. Similar to Park's method, Shu's method requires fast DCT-II and IDCT-II algorithms to efficiently resize images in the DCT domain.

Table 5.9 compares the computational cost of various transcoding methods for downsizing images by a factor of P/Q . For all downsizing factors, shu(II) is found to be more computation efficient as compared to shu(I). However, shu(II) is slower than the proposed and Park's methods for most downsizing factors. The constraints used in park(II) does not always lead to lower computational cost as compared to park(I) for downsizing factor P/Q such as $4/5$, $5/6$, and $6/7$. For these downsizing factors, park(I) and park(II) have the same computational cost. It is clear that $\text{prop}(N_{0.05})$ and $\text{prop}(N_{PT})$ are significantly faster to compute as compared to $\text{prop}(N)$. Among the various methods, $\text{prop}(N_{PT})$ is found to possess the lowest computational cost for all resizing factors listed in Table 5.9.

Table 5.9 Downsizing of images by a factor of P/Q using various methods.

Method	Computational cost per pixel (c_m, c_a)		
	$P/Q = 2/3$	$P/Q = 3/4$	$P/Q = 4/5$
prop(N)	(15.00, 17.78)	(28.88, 33.03)	(43.20, 49.50)
prop($N_{0.05}$)	(4.58, 6.39)	(5.60, 8.33)	(8.33, 11.86)
prop(N_{PT})	(2.04, 2.92)	(2.11, 2.84)	(3.28, 4.27)
park(I) [PP06]	(4.24, 12.85)	(4.05, 15.48)	(4.82, 13.73)
park(II) [PP06]	(2.15, 8.06)	(2.89, 8.26)	(4.82, 13.73)
shu(I) [SC07]	(12.01, 35.35)	(15.75, 19.25)	(21.60, 27.54)
shu(II) [SC07]	(4.24, 12.85)	(9.95, 13.34)	(15.75, 20.97)

Method	Computational cost per pixel (c_m, c_a)		
	$P/Q = 5/6$	$P/Q = 6/7$	$P/Q = 7/8$
prop(N)	(53.47, 61.79)	(65.00, 75.94)	(85.31, 97.62)
prop($N_{0.05}$)	(10.29, 15.50)	(6.96, 11.91)	(9.35, 15.35)
prop(N_{PT})	(4.04, 5.00)	(2.12, 3.24)	(2.59, 3.69)
park(I) [PP06]	(5.65, 18.52)	(11.81, 24.91)	(16.44, 32.55)
park(II) [PP06]	(5.65, 18.52)	(11.81, 24.91)	(6.47, 11.57)
shu(I) [SC07]	(29.79, 36.86)	(40.39, 49.05)	(49.22, 59.41)
shu(II) [SC07]	(21.96, 27.99)	(26.69, 34.95)	(32.81, 42.42)

Furthermore, the computational cost of prop(N), prop($N_{0.05}$), and prop(N_{PT}) is relatively balanced in terms of addition and multiplication as compared to Park's and Shu's methods. This is due to the fact that Park's and Shu's methods require fast DCT-II and IDCT-II algorithms [LLM89, CS93, BY98], which generally leads to a computational cost requiring more additions as compared to multiplications.

Table 5.10 and 5.11 summarize the PSNR values of the images downsized by factors of $2/3$ and $3/4$, respectively. Similar to the proposed method, the PSNR values of the Park's and Shu's methods are obtained from the original test-images and resized images, which are obtained from their optimal least square upsizing

Table 5.10 PSNR values for downsizing by a factor of 2/3 using various methods.

Test Image	PSNR (dB)						
	prop($\tilde{N}$)			Park		Shu	
	$\tilde{N} = N$	$\tilde{N} = N_{0.05}$	$\tilde{N} = N_{PT}$	park(I)	park(II)	shu(I)	shu(II)
TI-1	24.45	24.45	24.43	24.43	24.27	24.45	24.42
TI-2	28.87	28.87	28.84	28.86	28.78	28.87	28.84
TI-3	32.38	32.38	32.26	32.37	32.19	32.38	32.36
TI-4	33.70	33.70	33.56	33.71	33.51	33.70	33.67
TI-5	35.93	35.93	35.83	36.01	35.88	35.93	35.89
TI-6	38.74	38.74	38.36	38.76	38.59	38.74	38.71
TI-7	38.19	38.19	38.04	38.27	38.17	38.19	38.16
TI-8	38.47	38.47	38.09	38.48	38.32	38.47	38.45
TI-9	36.20	36.20	36.10	36.29	36.13	36.20	36.17
TI-10	30.29	30.29	30.21	30.26	30.05	30.29	30.26
TI-11	31.01	31.01	30.92	31.01	30.84	31.01	30.99
TI-12	33.66	33.66	33.49	33.68	33.54	33.66	33.63
TI-13	25.73	25.73	25.69	25.73	25.54	25.73	25.70
TI-14	35.31	35.31	35.09	35.32	35.14	35.31	35.28
TI-15	28.56	28.56	28.49	28.55	28.37	28.56	28.53
TI-16	32.61	32.61	32.51	32.60	32.42	32.61	32.58
TI-17	30.04	30.04	29.97	30.04	29.87	30.04	30.01
TI-18	27.15	27.15	27.12	27.15	26.97	27.15	27.12
TI-19	34.08	34.08	33.89	34.09	33.96	34.08	34.06
TI-20	39.11	39.11	38.97	39.35	39.24	39.11	39.07
Mean	32.72	32.72	32.59	32.75	32.59	32.72	32.69

filters. For Shu's and Park's methods, the optimal least square upsizing filters are obtained from shu(I) and park(I), respectively. The various methods discussed in Table 5.10 and 5.11 are found to produce similar PSNR values for resizing TI-1 to TI-20. Among the seven transcoding methods, prop(N_{PT}) and park(II) generally produce the resized images having lowest PSNR values.

From Table 5.10 and 5.11, we observe negligible PSNR differences between the downsized images obtained using prop(N) and prop($N_{0.05}$). Small PSNR difference is observed between prop(N) and prop(N_{PT}), and our subjective evaluation revealed that the downsized images from prop(N) and prop($N_{0.05}$) are visually identical. In fact, it is concluded from our subjective evaluation that

Table 5.11 PSNR values for downsizing by a factor of 3/4 using various methods.

Test Image	PSNR (dB)						
	prop($\tilde{N}$)			Park		Shu	
	$\tilde{N} = N$	$\tilde{N} = N_{0.05}$	$\tilde{N} = N_{PT}$	park(I)	park(II)	shu(I)	shu(II)
TI-1	26.21	26.21	26.16	26.24	25.42	26.21	26.20
TI-2	31.41	31.41	31.32	31.54	31.02	31.41	31.40
TI-3	33.94	33.94	33.71	33.96	33.16	33.94	33.93
TI-4	35.15	35.15	34.90	35.20	34.38	35.15	35.15
TI-5	37.05	37.05	36.86	37.30	36.67	37.05	37.04
TI-6	40.62	40.62	39.97	40.71	39.78	40.62	40.61
TI-7	40.44	40.44	40.12	40.67	40.00	40.44	40.43
TI-8	39.87	39.87	39.27	39.90	39.22	39.87	39.87
TI-9	38.00	38.00	37.76	38.23	37.42	38.00	37.99
TI-10	32.00	32.00	31.83	32.05	31.02	32.00	31.99
TI-11	32.78	32.78	32.58	32.82	32.01	32.78	32.77
TI-12	35.20	35.20	34.90	35.26	34.54	35.20	35.19
TI-13	27.01	27.01	26.93	27.02	26.22	27.01	27.00
TI-14	37.18	37.18	36.76	37.26	36.38	37.18	37.17
TI-15	29.96	29.96	29.83	29.97	29.17	29.96	29.95
TI-16	34.49	34.49	34.28	34.54	33.76	34.49	34.48
TI-17	31.73	31.73	31.58	31.76	30.88	31.73	31.72
TI-18	28.72	28.72	28.64	28.79	27.99	28.72	28.71
TI-19	35.06	35.06	34.75	35.07	34.45	35.06	35.05
TI-20	40.22	40.22	39.94	40.71	40.08	40.22	40.20
Mean	34.35	34.32	34.10	34.45	33.68	34.35	34.34

the downsized images from the seven methods in Tables 5.10 and 5.11 are visually similar.

Table 5.12 presents the average PSNR values for downsizing images by a factor of P/Q , where $P/Q = 2/3, 3/4, 4/5, 5/6, 6/7$, and $7/8$. For these downsizing factors, the PSNR values of the downsized images obtained from $\text{prop}(N)$ and $\text{prop}(N_{0.05})$ are very similar to those from park(I) , shu(I) , and shu(II) . For downsizing factor $P/Q = 2/3, 3/4$, and $7/8$, PSNR values of the downsized images from $\text{prop}(N_{0.05})$ are higher than park(II) , especially in the case of $P/Q = 7/8$. For the two Shu's methods, the PSNR values of the downsized images are almost identical. For $P/Q = 4/5, 5/6$, and $6/7$, park(II) is the same as park(I) , hence, the same PSNR values are found in park(I) and park(II) for these downsizing factors.

Table 5.12 Average PSNR values for downsizing images using various methods.

P/Q	Average PSNR (dB)						
	$\text{prop}(\tilde{N})$			Park		Shu	
	$\tilde{N} = N$	$\tilde{N} = N_{0.05}$	$\tilde{N} = N_{\text{PT}}$	park(I)	park(II)	shu(I)	shu(II)
2/3	32.72	32.72	32.59	32.75	32.59	32.72	32.69
3/4	34.35	34.32	34.10	34.45	33.68	34.35	34.34
4/5	35.38	35.38	34.98	35.58	35.58	35.38	35.38
5/6	36.29	36.29	35.82	36.47	36.47	36.29	36.29
6/7	37.11	37.06	36.56	37.27	37.27	37.11	37.11
7/8	38.07	38.03	37.31	38.20	36.11	38.07	38.07

The PSNR and MSSIM values of the test-images (TI-1 to TI-45) resized by downsizing factor P/Q , where $P/Q = 2/3, 3/4, 4/5, 5/6, 6/7$, and $7/8$ are summarized in Appendix C.

Next, different downsizing factors are applied in the horizontal and vertical directions. This downsizing factor is expressed as $P/Q \times R/S$, where the downsizing factors P/Q and R/S are applied vertically and horizontally, respectively. Such downsizing factor is commonly used in applications where HD image and video contents having resolution of 1920×1080 are downsized for mobile devices. Downsized image and video contents of resolutions 720×480 , 640×480 and 320×240 are of interest since many mobile devices have displays with such resolutions. These resolutions translate to downsizing factors of $4/9 \times 3/8$, $4/9 \times 1/3$ and $2/9 \times 1/6$, respectively.

Table 5.13 presents a PSNR comparison of seven transcoding methods to downsize images by a factor of $P/Q \times R/S$. In this comparison, a series of HD test images are used. These HD test images are extracted from [URL01], and are shown in Fig. 5.13. For all downsizing factors listed in Table 5.13, the PSNR values of the resized images obtained using $\text{prop}(N)$ and $\text{prop}(N_{0.05})$ are similar to those obtained using park(I) , shu(I) and shu(II) . For most downsizing factors,

Fig. 5.13 High definition grayscale test-images (1920×1080) in (a) to (c) are referred as HTI-1 to HTI-3, respectively.

Table 5.13 PSNR values for downsizing by a factor of $P/Q \times R/S$.

Test Image	PSNR (dB) for resizing by $4/9 \times 3/8$						
	prop($\tilde{N}$)			Park		Shu	
	$\tilde{N} = N$	$\tilde{N} = N_{0.05}$	$\tilde{N} = N_{PT}$	park(I)	park(II)	shu(I)	shu(II)
HTI-1	39.83	39.81	39.24	39.76	38.85	39.83	39.83
HTI-2	40.35	40.34	40.04	40.35	40.09	40.35	40.35
HTI-3	38.85	38.83	37.59	38.78	37.96	38.85	38.85
Avg.	39.68	39.66	38.96	39.63	38.97	39.68	39.68
Test Image	PSNR (dB) for resizing by $4/9 \times 1/3$						
	prop($\tilde{N}$)			Park		Shu	
	$\tilde{N} = N$	$\tilde{N} = N_{0.05}$	$\tilde{N} = N_{PT}$	park(I)	park(II)	shu(I)	shu(II)
HTI-1	38.87	38.87	33.69	38.84	38.56	38.87	38.87
HTI-2	39.99	39.99	36.73	39.99	39.89	39.99	39.98
HTI-3	37.73	37.73	32.36	37.69	37.45	37.73	37.72
Avg.	38.86	38.86	34.26	38.84	38.63	38.86	38.86
Test Image	PSNR (dB) for resizing by $2/9 \times 1/6$						
	prop($\tilde{N}$)			Park		Shu	
	$\tilde{N} = N$	$\tilde{N} = N_{0.05}$	$\tilde{N} = N_{PT}$	park(I)	park(II)	shu(I)	shu(II)
HTI-1	31.12	31.05	31.05	31.10	30.94	31.12	31.12
HTI-2	35.69	35.69	35.64	35.68	35.51	35.69	35.69
HTI-3	30.19	30.19	30.13	30.18	30.05	30.19	30.19
Avg.	32.33	32.28	32.27	32.32	32.17	32.33	32.33

prop(N_{PT}) has the lowest PSNR values, but subjective evaluation revealed that the downsized images are visually identical to those from other methods. shu(I) and shu(II) yield negligible PSNR difference, and perform similarly to prop(N) and prop($N_{0.05}$) in terms of PSNR values. The PSNR values of prop(N_{PT}) are

found to be similar to those of $\text{prop}(N_{0.05})$, park(I) , shu(I) and shu(II) , except for resizing by $4/9 \times 1/3$. Among the three HD images, downsizing HTI-3 using $\text{prop}(N_{\text{PT}})$ results in significant PSNR difference, and downsizing HTI-3 by a factor of $4/9 \times 1/3$ yields the highest PSNR difference of 5.37 dB. This is attributed to the large amount of high spatial frequency content in HTI-3, which is discarded in the down-sampling stage due to the higher visibility threshold at higher spatial frequencies. This operation led to the downsized image having the lowest PSNR value in Table 5.13. Again, it should be noted that the downsized images from the various methods are visually identical from our subjective evaluation.

Table 5.14 compares the computational cost of the proposed, Park's and Shu's transcoding methods. The computational cost of DCT-II and IDCT-II algorithms of various lengths are obtained from [LLM89, CS93, BY98] to derive the computational cost of Park's and Shu's transcoding methods. From Table 5.14, it is clear that $\text{prop}(N_{\text{PT}})$ possesses one of the lowest computational cost among all the transcoding methods discussed in this chapter. In fact, $\text{prop}(N_{\text{PT}})$ achieves computational cost reduction up to 97% as compared to $\text{prop}(N)$. For all downsizing factors in Table 5.14, park(II) and shu(II) have lower computational cost as compared to park(I) and shu(I) , respectively. Also, park(II) has the second lowest computational cost for most downsizing factors, and requires the lowest multiplications for the downsizing factor $4/9 \times 1/3$. shu(II) is more efficient than $\text{prop}(N)$, but is of higher complexity than $\text{prop}(N_{0.05})$, $\text{prop}(N_{\text{PT}})$, and park(II) for all downsizing factors. Similar to the observations in Table 5.9, the

Table 5.14 Computational cost for downsizing by a factor of $P/Q \times R/S$.

Method	Computational cost per pixel (c_m, c_a)		
	$4/9 \times 3/8$	$4/9 \times 1/3$	$2/9 \times 1/6$
prop(N)	(33.56, 38.58)	(26.67, 30.81)	(11.41, 13.81)
prop($N_{0.05}$)	(1.33, 4.72)	(5.94, 9.10)	(0.53, 1.96)
prop(N_{PT})	(0.99, 1.58)	(3.03, 3.84)	(0.28, 0.58)
park(I)	(4.04, 13.51)	(4.12, 12.92)	(4.13, 12.44)
park(II)	(1.58, 5.38)	(1.49, 5.41)	(0.44, 1.48)
shu(I)	(24.24, 81.38)	(19.78, 65.79)	(9.44, 28.85)
shu(II)	(10.08, 30.57)	(8.36, 24.81)	(3.67, 11.13)

computational cost of prop(N_{PT}) is relatively balanced between multiplications and additions.

5.5 Conclusions

A method to arbitrarily resize images in the DCT domain was presented in this chapter. This transcoding method is based on the spatial relationship of DCT-II coefficients between a block and its sub-block. The generalization of an observation of this spatial relationship leads to a sparse matrix representation, which facilitates significant reduction of the computational cost of our proposed transcoding method.

Generally speaking, resizing of images by a factor of P/Q is obtained by a two-stage structure, which involve up-sampling and down-sampling of images by factors of P and Q , respectively. Choosing to perform up-sampling prior to down-sampling produces visually better images, but such an approach requires high computational cost. By utilizing the sparse matrix representation of $\mathbf{D}_{l_1, l_2}$ and $\mathbf{U}_{l_1, l_2}$ in the proposed transcoding method prop(N), significant reduction of

computational cost is achieved. To further reduce the computational cost of the proposed transcoding method, resizing of images is realized by using a subset of up-sampled DCT-II coefficients in the down-sampling stage. By analyzing the reduction of PSNR with respect to the number of up-sampled DCT-II coefficients used in the down-sampling stage and the downsizing factor P/Q , we empirically determined the lowest number of up-sampled DCT-II coefficients that can be used in the down-sampling stage without degrading the visual quality of the downsized images. Hence, an efficient method $\text{prop}(N_{0.05})$ that produces downsized images with good visual quality was obtained.

An alternative and more effective approach to reduce the computational cost of the proposed transcoding method was discussed in Section 5.3. The base visibility threshold derived in Chapter 4 was employed to identify perceptually significant up-sampled DCT-II coefficients to be used in the down-sampling stage, and led to a perceptually-tuned transcoding method $\text{prop}(N_{\text{PT}})$. This perceptually-tuned transcoding method produces visually identical downsized images as $\text{prop}(N)$, which was verified by subjective evaluation and perceptual distortion score. $\text{prop}(N_{\text{PT}})$ achieves up to 97% computational cost reduction as compared to $\text{prop}(N)$, and is found to be the most computation efficient among the methods discussed for most downsizing factors.

Chapter 6

Conclusions and Future Works

6.1 Conclusions

This thesis first considered the derivation and analysis of the computational model for spatial JND. By integrating the computational model for JND with the SBDCT, a perceptually-tuned image compression system based on the SBDCT is achieved. Subsequently, this thesis derived an efficient perceptually-tuned transcoding method to arbitrarily resize images by a factor of P/Q .

Generally speaking, most computational models for spatial JND consider three properties of the HVS, namely, CSF, luminance adaptation, and intra- and inter-band contrast masking. An effective computational model for JND should not only provide a good estimation of the limitations of the HVS, but also take into account of the imagery system that the JND model is applied into. On this note, the computational model for JND should consider the transform algorithm of an image compression system, such as DCT and DWT in JPEG [ISO94] and JPEG 2000 [ISO00], respectively. However, many PICs only considered a perceptually-tuned quantization stage to embed quantization errors into compressed images [SJ89, Wat93, CL95, ZLX05]. The primary objective of such PICs is to increase compression ratio while minimizing the perceptible distortion in the compressed images. In such PICs, there is no further attempt to integrate the computational model for JND and the computational structure for the transform algorithm.

As mentioned earlier, this thesis discussed an image compression system using SBDCT instead of the DCT-II. It has been shown by Jung *et al.* [JMM96] that the

computational cost of DCT-II is reduced by approximating the DCT-II with SBDCT. Even though SBDCT is derived from DCT-II, the computational structure for SBDCT is quite different from DCT-II. SBDCT operates on subbands of an $N \times N$ input sequence, instead of an $N \times N$ input sequence in the case of DCT-II. Apart from that the computational structure for SBDCT is well-suited for parallel implementation, its computational structure also led to an efficient approximation of the DCT-II. By only using the low-low band of an $N \times N$ input sequence, ApD1 yields a good approximation of the DCT-II if the $N \times N$ input sequence of the DCT-II is highly correlated. This is attributed to the fact that the low-low band of an $N \times N$ input sequence contains most of the energy for a highly correlated $N \times N$ input sequence. Therefore, it is clear that the performance of ApD1 degrades significantly at edge and texture regions of an image, and resulting in perceptible distortion at these regions. As discussed in Section 3.3, significant aliased component is found with ApD1. These aliased component is primarily caused from the omission of the rest of the subbands, which may contain considerable amounts of energy. To mitigate this problem, a total of seven approximations (ApD1 to ApD7) of the DCT-II based on SBDCT were proposed in Section 3.2.2.

Based on the energy distribution of the subbands of an $N \times N$ input sequence, AdS2 adaptively selects a suitable approximation of the DCT-II from ApD1 to ApD7 was proposed in Section 3.3. Based on this idea, a computational model for JND was integrated with SBDCT, and this algorithm is referred as PSBDCT. Apart from deriving a perceptually-tuned quantization stage for SBDCT using the computational model for JND, the JND model was also used to select the perceptually significant subbands of each $N \times N$ sub-image. Note that $N \times N$ input

sequence is replaced by $N \times N$ sub-image when considering SBDCT and DCT-II for image processing. As shown in Table 4.10, it is clear that the average computational cost of PSBDCT is significantly reduced as compared to the fullband SBDCT, especially for TI-6, TI-7, TI-8, and TI-20.

In [ZLX05, ZLX08, WN09], block classification of $N \times N$ sub-images of an image is used to improve the estimation of contrast masking. From our simulations presented in Section 4.1.2, pixel-based block classification possesses superior performance as compared to DCT-based block classification. By approximating the edge map of an image using the subbands $\mathbf{c}_{1,n}$, $\mathbf{c}_{2,n}$, and $\mathbf{c}_{3,n}$ of an $N \times N$ sub-image, the highly complex Canny edge operator used in Wei's block classification [WN09] is avoided by our block classification. On the other hand, Zhang [ZLX05, ZLX08] performed block classification of each $N \times N$ input sequence in the DCT domain. Zhang's block classification requires seven thresholds as seen in Table 2.1 and (2.31), whereas only three thresholds are required by the proposed PIC as shown in (4.6) and (4.7). These thresholds in (4.6) and (4.7) are related to the pixel value and the average number of edge pixels in each $N \times N$ sub-image, respectively. Hence, these thresholds are relatively independent of N , but this is not the case for the thresholds used in Zhang's block classification. On top of above advantages of the proposed block classification, our simulations also revealed that proposed block classification produces more consistent results as compared to Wei's and Zhang's block classification.

As reported in [ZLX08], JND threshold computed from pixels [CL95, CC96, CB99, YLL03, YLL05, LLP10] generally do not account for CSF, hence, PICs with pixel-based JND models are unable to discern noise with respect to the spatial frequency sensitivity of the HVS. As discussed in Section 4.1.5, the

proposed PIC did not directly apply a pixel-based JND model to determine the quantization matrix. In the case of the proposed PIC, the pixel-based JND model is first converted into a subband JND model, and the subband JND model is then used to locally adapt the JND threshold for $N \times N$ DCT-II coefficients [CR90]. From our subjective evaluation, the proposed PIC was able to take advantage of both pixel- and subband-based JND models to obtain the best visual quality for images compressed within 0.4 – 0.5 bpp, as observed from Table 4.6 and 4.9 as well as Fig. 4.17 to 4.19. From our discussion presented in Section 4.3, the perceptual distortion score from the proposed PIC is also found to have the best correlation with human viewing.

With the proposed PIC developed around the SBDCT, it is desirable to maintain the computational cost of SBDCT as low as possible. One of the key challenges is to reduce the computational cost associated with the $N \times N$ twiddle factor matrices $\mathbf{W}_{cc}$, $\mathbf{W}_{cs}$, $\mathbf{W}_{sc}$, and $\mathbf{W}_{ss}$ defined in (3.12). Directly computing these matrices in (3.12) requires a total of 256 multiplications. Several steps had been undertaken to optimize the computational cost of SBDCT, namely, (i) exploiting a 2-D symmetry in the SBDCT coefficients as well as zero coefficients; (ii) established the link between $N/2 \times N/2$ fast 2-D DCT-II algorithm and the subband transforms of SBDCT defined in (3.6); (iii) implementation of the $N/2 \times N/2$ fast 2-D DCT-II algorithm using the direct 2-D DCT-II computation based on Huang's computational structure for the DCT-II [HWH98, HW99]; and (iv) factorization of (3.12) to further reduce the computational cost of the twiddle factor matrices in SBDCT.

By factorizing out the twiddle factor matrices from its associated subband transforms in SBDCT, a 2-D symmetry was found in the four subband transforms

of SBDCT, as given in (3.13) to (3.16). This 2-D symmetry of the SBDCT coefficients ensures that the number of unique SBDCT coefficients (before scaling by the twiddle factor matrices) remains the same as the DCT-II. By investigating the subband transforms of SBDCT, it was found that the subband transforms $\mathbf{C}_{1,n}^{2D}$, $\mathbf{C}_{2,n}^{2D}$, $\mathbf{C}_{3,n}^{2D}$ possess similar computational structure as the subband transform $\mathbf{C}_{0,n}^{2D}$, which is the $N/2 \times N/2$ DCT-II. Using permutation matrix $\mathbf{P}_3$, we computed $\mathbf{C}_{1,n}^{2D}$, $\mathbf{C}_{2,n}^{2D}$, $\mathbf{C}_{3,n}^{2D}$ using the same computational structure for $\mathbf{C}_{0,n}^{2D}$. Therefore, the task of optimizing the computational structures for $\mathbf{C}_{0,n}^{2D}$, $\mathbf{C}_{1,n}^{2D}$, $\mathbf{C}_{2,n}^{2D}$, and $\mathbf{C}_{3,n}^{2D}$ becomes the sole task of optimizing the computational structure of $\mathbf{C}_{0,n}^{2D}$. Having this in mind, a fast computational structure for an $N/2 \times N/2$ DCT-II was derived in Section 3.2.1, and this computational structure forms the basis of the FSA that was extensively used in Chapter 3 and 4. This fast computational structure for an $N/2 \times N/2$ DCT-II is based on the direct 2-D DCT-II computation proposed by Huang *et al.* [HWH98, HW99], which requires significant lesser computational as compared to the traditional row-column decomposition method used in [LLM89, JMM96, HH04].

Advantages of Huang's computational structure include high regularity as well as low computational cost. Huang *et al.* implemented their fast 2-D DCT-II using N N -point DCT-II, thus, the computational cost of their computational structure becomes directly proportional to the 1-D DCT-II algorithm used in the computational structure for the 2-D DCT-II. From this discussion, it is obvious that the optimal computational cost of Huang's computational structure is only achieved when the 1-D DCT-II possesses optimal computational cost. In this thesis, a 4-point DCT-II was derived from a 8-point DCT-II structure proposed in

[LLM89, TGW05b]. This 4-point DCT-II achieves the lower theoretical bound of multiplications for the N -point DCT-II [HB85, DH87, WF92]. Using this computational structure for the 4-point DCT-II, we obtained a 4×4 DCT-II algorithm having a computational cost of (16, 74), which also achieved the lower theoretical bound of multiplications for the $N \times N$ DCT-II [FW92].

The above observations led to the successful development of an $N \times N$ FSA, which requires (204, 536) for $N = 8$. As compared to the computational cost of (496, 544) required by the FSA reported in [JMM96], the proposed FSA achieves up to 58% reduction in terms of multiplications. This computational structure for SBDCT also led to seven approximations ApD1 to ApD7 of the 2-D DCT-II algorithm, which inherit the efficient computation of the FSA. As discussed in Section 3.2.2, these approximations were obtained by using different combinations of the subband transforms of the SBDCT. By selecting ApD1 to ApD7 based on the energy distribution of each $N \times N$ sub-image, vast improvement in the image quality of the reconstructed images is obtained as compared to the approximation of DCT-II proposed in [JMM96, HH04]. With these results, it becomes natural to formulate a method to select ApD1 to ApD7 to process each $N \times N$ sub-image with respect to the HVS as the next stage of enhancement. Eventually, this step led to the development of a perceptually-tuned SBDCT image coder that was derived and analyzed in Chapter 4.

Based on the spatial relationship of DCT-II coefficients between a block and its sub-block, and the observations from SBDCT, an efficient image transcoding method was proposed in Chapter 5. The generalization of a symmetry property of the spatial relationship of the DCT-II coefficients led to a sparse matrix representation, which forms the basis of the proposed image transcoding method

to resize images. Generally speaking, resizing of images by a factor of P/Q can be divided into two stages, which include up-sampling and down-sampling of images by factors of P and Q , respectively. By selectively discarding some up-sampled DCT-II coefficients in the down-sampling stage, it was found that the computational cost of resizing images by a factor of P/Q is tremendously reduced without introducing perceptible distortion in the resized images. This observation was backed up by the simulations and subjective evaluation discussed in Section 5.2.3.

Using standard results of multirate signal processing, the equivalent filter bank structure of the down-sampling stage of the proposed transcoding method was obtained and analyzed. From visual inspection of Fig. 5.6, it is clear that the equivalent filter response of the down-sampling stage does not change significantly for all values of $\tilde{N}$, where $\tilde{N}$ corresponds to different number of up-sampled DCT-II coefficients used in the down-sampling stage. In particular, the filter response F_0 almost remains unchanged for $5 \leq \tilde{N} \leq 8$ for $P/Q = 2/3$ and $3/4$. This observation warranted more investigation to determine the relation between P , Q , and $\tilde{N}$. This relation is important as it can be used to reduce the computational cost of the proposed transcoding method, and the related analysis is reported in Section 5.2.3.

Analyzing the plots presented in Fig. 5.8 to 5.10, several observations were made. First, the reduction of $\text{ADP}(\tilde{N})$ diminishes when Q and $\tilde{N}$ increase. Second, increasing $\tilde{N}$ yields smaller values of $\text{ADP}(\tilde{N})$ for all values of P . Third, $\text{ADP}(\tilde{N})$ increases when P is increased. Thus, it is concluded that there exists a direct relation between P and $\text{ADP}(\tilde{N})$, and an inverse relation between

Q and $\text{ADP}(\tilde{N})$. These observations together with Fig. 5.8 to 5.10 led to the selection of the block size $\tilde{N}_{0.05}$ of the proposed transcoding method, which produces visually good resized images at a reduced computational cost as compared to the proposed transcoding method with block size N .

An extension of the proposed transcoding method was explored in Section 5.3. This extension applied the base visibility threshold derived in Section 4.1.4 to determine the number of up-sampled DCT coefficients to be used in the down-sampling stage. This extension gave rise to a perceptually-tuned transcoding method $\text{prop}(N_{\text{PT}})$. From our subjective evaluation, $\text{prop}(N_{\text{PT}})$ produces visually identical downsized images as those produced by $\text{prop}(N)$, but $\text{prop}(N_{\text{PT}})$ attained up to 96% reduction of the computational cost required by $\text{prop}(N)$ when resizing images by a factor of P/Q . As compared to Park's and Shu's image transcoding methods, $\text{prop}(N_{\text{PT}})$ was also found to have the lowest computational cost to resize images for almost all the resizing factors discussed in Chapter 5. In addition, the proposed transcoding method do not require fast DCT-II or IDCT-II algorithms, which is not the case for Shu's and Park's transcoding methods. Due to this reason, their methods will not be computational efficient if there is no fast DCT-II or IDCT-II algorithm available for their computation. Also, fast DCT-II and IDCT-II algorithms [LLM89, WP89a, WP89b, DG90, CH91, CL91, CYL91, WP91, CL92, FW92, CL93, PD96, HWH98, HW99, BR00] tend to require more additions as compared to multiplications, and results in a unbalanced computational cost. Since the proposed transcoding method does not require any fast DCT-II and IDCT-II algorithm, it is found to have a relatively

balanced computational cost as compared to the other transcoding methods discussed in Chapter 5.

6.2 Future Works

The ultimate receiver for most imagery systems is human, therefore, it is highly desirable to account for the characteristics and limitations of the HVS into such systems to achieve its optimal performance in the perceptual sense.

One of the main challenges is to correlate the objective score with the subjective score of a human. Extensive efforts were undertaken by VQEG to develop a systematic approach to validate an objective video quality model. In the methodology used by VQEG [VQE03], Pearson correlation and Spearman rank-order correlation are used to rank various video quality models under examination. However, it should not be inferred that Pearson correlation and Spearman rank-order correlation are the best statistics to describe the relation between the objective score from the various video quality models and the subjective rating from a human viewer. Since this relation between the objective score and the subjective rating need not be linear, questions remain on the validity of applying Pearson correlation to measure the statistics of data from such a model. On this note, it may be worthwhile to investigate and determine a statistic measure that is more well-suited to model the nonlinearity in the relation of (if any) the scores between a video quality model and subjective viewing.

One of the key parameters of a computational model for JND is the viewing distance of an image under examination. This thesis considered several of the pixel-based JND models from [CL95, YLL03, YLL05], which defined the viewing distance at six times of the image height. It would be interesting to analyze the behaviour of the JND threshold with respect to the viewing distance,

and determine the relation between the JND threshold and the viewing distance. As the computational cost of most JND models tend to be high, such a relation can be employed to provide a good estimate of JND threshold at different viewing distances. Another possible extension of the proposed computational model for JND is to consider the temporal effects of the HVS. Robson [Rob66] found that the temporal CSF has a characteristic of a bandpass filter at 0.5 cpd, and the temporal CSF follows the shape of a low pass filter at higher spatial frequencies. By incorporating a spatio-temporal computational model into a video compression system, we could integrate motion compensation from perceptually-tuned video frames with the selection of appropriate approximated DCT-II algorithm, aiming to reduce computational cost and improving video compression with respect to the HVS.

A possible refinement of the perceptually-tuned image transcoding method is to establish a relation between the HVS and the equivalent filter banks of the up-sampling and down-sampling stages. The key objective of this refinement is to effectively reduce ringing and blurring artifacts by guiding the selection of upsizing and downsizing filters to resize image or video contents with respect to HVS.

One obvious extension of the proposed PIC is the modeling of the JND threshold for color images since most images are color images rather than grayscale images. The adaptation of (4.17) to chrominance components has been explored in [PAW93b], which applied the results on the differences in the CSF curves for the luminance and chrominance curves reported by [Pet92].

Author's Publications

- [TGW05a] E. L. Tan, W. S. Gan, and M. T. Wong, "Perceptually lossless coder based on just noticeable distortion profile with subband DCT," in *Proc. Int'l Symp. Consumer Electronics*, pp. 253-257, 2005.
- [TGW05b] E. L. Tan, W. S. Gan, and M. T. Wong, "Practical implementation of subband DCT based coder," in *Proc. Int'l Symp. Intelligent Signal Process. Commun. Systems*, pp. 485-488, 2005.
- [TGC07] E. L. Tan, W. S. Gan, Y. K. Chong, and M. T. Wong, "Perceptually tuned subband coder with region of interest," *Int'l Conf. Information, Commun. & Signal Process.*, pp. 1-5, 2007.
- [TGW07] E. L. Tan, W. S. Gan, and M. T. Wong, "Fast Arbitrary Resizing of Images in DCT Domain," *Int'l Conf. Multimedia and Expo*, pp. 1671-1674, 2007.
- [TGW08] E. L. Tan, W. S. Gan, and M. T. Wong, "Fast image resizing in discrete cosine transform domain with spatial relationship between DCT block and its sub-blocks," *Int'l Conf. Audio, Language and Image Process.*, pp. 1683-1687, 2008.
- [TG11] E. L. Tan and W. S. Gan, "Perceptually tuned subband coder for JPEG," *J. Real-Time Image Process.*, vol. 6, no. 2 pp. 101-115, 2011.
- [TG11a] E. L. Tan and W. S. Gan, "Perceptually-tuned subband discrete cosine transform coder," *Submitted to IEEE Trans. Circuits Syst. Video Technol.*, 2011.
- [TGM11] E. L. Tan, W. S. Gan, and S. K. Mitra, "Fast arbitrary resizing of images in the discrete cosine transform domain," *IET Image Process.*, vol. 5, no. 1, pp. 73-86, 2011.

- [YTG07] Y. Yang, E. L. Tan, W. S. Gan, and Y. K. Chong, "Scalable image coder for mobile device," *Int'l Conf. Information, Commun. & Signal Process.*, pp. 1-5, 2007.

Bibliography

- [AS69] V. R. Algazi and D. J. Sakrison, "On the optimality of the Karhunen-Loeve expansions," *IEEE Trans. Inform. Theory*, vol. 15, no. 2, pp. 319-321, March 1969.
- [ANR74] N. Ahmed, T. Natarajan and K. R. Rao, "Discrete cosine transform," *IEEE Trans. Computers*, vol. C-23, no. 1, pp. 90-93, January 1974.
- [AP92] A. J. Ahumada Jr. and H. A. Peterson. "Luminance-Model-Based DCT Quantization for Color Image Compression," in *Human Vision, Visual Processing, and Digital Display III*, B. E. Rogowitz, ed. (Proceedings of the SPIE, 1992).
- [BC69] C. Blakemore and F. W. Campbell, "On the existence of neurones in the human visual system selectivity to the orientation and size of retinal images," *J. Physiology*, vol. 203, no. 1, pp. 237-260, July 1969.
- [BKW75] M. A. Berkley, F. Kitterle, and D. W. Watkins, "Grating visibility as a function of orientation and retinal eccentricity," *Vision Res.*, vol. 15, no. 2, pp. 239-244, February 1975.
- [BGS82] C. Braccini, G. Gambaradella, G. Sandini, and V. Tagliasco, "A model of the early stages of the human visual system: functional and topological transformations performed in the peripheral visual field," *Biological Cybernetics*, vol. 44, no. 1, pp. 47-48, May 1982.
- [BK97] V. Bhaskaran and K. Konstantinides, *Image and video compression standards: algorithms and architectures*, Kluwer Academic Publishers, 1997.
- [BY98] G. Bi and L. W. Yu, "DCT algorithms for composite sequence length," *IEEE Trans. Signal Process.*, vol. 46, no. 3, pp. 554-562, March 1998.

- [BR00] V. Britanak and K. R. Rao, "Two-dimensional DCT/DST universal computational structure for $2m \times 2n$ block sizes," *IEEE Trans. Signal Process.*, vol. 48, no. 11, pp. 3250-3255, November 2000.
- [Bov05] A. C. Bovik, *Handbook of image and video processing (communication, networking and multimedia)*, Academic Press, 2005.
- [CR68] F. W. Campbell and J. G. Robson, "Application of Fourier analysis to the visibility of gratings," *J. Physiology*, vol. 197, no. 3, pp. 551-566, August 1968.
- [Can86] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679-698, November 1986.
- [Cor90] T. N. Cornsweet, *Visual Perception*, Academic Press, 1990.
- [CR90] B. Chitprasert and K.R. Rao, "Human visual weighted progressive image transmission," *IEEE Trans. Commun.*, vol. 38, no. 7, pp. 1040-1044, July 1990.
- [CH91] S. C. Chan and K. L. Ho, "A new two-dimensional fast cosine transform algorithm," *IEEE Trans. Signal Process.*, vol. 39, no. 2, pp. 481-485, February 1991.
- [CL91] N. I. Cho and S. U. Lee, "Fast algorithm and implementation of 2-D discrete cosine transform," *IEEE Trans. Circuits Syst.*, vol. 38, no. 3, pp. 297-305, March 1991.
- [CYL91] N. I. Cho, I. D. Yun, and S. U. Lee, "A fast algorithm for 2-D DCT", *IEEE Int. Conf. Acoustic, Speech, Signal Process.*, pp. 2197-2200, May 1991.

- [CL92] N. I. Cho and S. U. Lee, "A fast 4×4 DCT algorithm for the recursive 2-D DCT," *IEEE Trans. Signal Process.*, vol. 40, no. 9, pp. 2166-2173, September 1992.
- [CL93] N. I. Cho and S. U. Lee, "On the regular structure for the fast 2-D DCT algorithm," *IEEE Trans. Circuits Syst.-II: Analog and Digital Signal Processing*, vol. 40, no. 4, pp. 259-266, April 1993.
- [CS93] Y. H. Chan and W. C. Siu, "Mixed-radix discrete cosine transform," *IEEE Trans. Signal Process.*, vol. 41, pp. 3157-3161, November 1993.
- [CL95] C. H. Chou and Y. C. Li, "A perceptually tuned subband image coder based on the measure of just-noticeable-distortion profile," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 5, no. 6, pp. 467-476, December 1995.
- [CC96] C. H. Chou and C. W. Chen, "A perceptually optimized 3-D subband codec for video communication over wireless channels," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 2, pp. 143-156, April 1996.
- [CB99] Y. J. Chiu and T. Berger, "A software-only videocodec using pixelwise conditional differential replenishment and perceptual enhancement," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 3, pp. 438-450, April 1999.
- [CE00] D. Santa-Cruz and T. Ebrahimi, "A study of JPEG 2000 still image coding versus other standards," *Proc. X European Signal Processing Conf.*, vol. 2, pp. 673-676, September 2000.
- [CSE00] C. Christopoulos, A. Skodras, and T. Ebrahimi, "The JPEG2000 still image coding system: an overview," *IEEE Trans. Consumer Electronics*, vol. 46, no. 4, pp. 1103-1127, November 2000.

- [DR58] W. D. Davenport and W. L. Root, *An introduction to the theory of random signals and noise*, McGraw-Hill, 1958.
- [Dau80] J. G. Daugman, "Two-dimensional spectral analysis of cortical receptive field profiles," *Vision Res.*, vol. 20, no. 10, pp. 847-856, 1980.
- [DAT82] R. L. DeValois, D. G. Albrecht, and L. G. Thorell, "Spatial frequency selectivity of cells in the macaque visual cortex," *Vision Res.*, vol. 22, no. 5, pp. 545-559, 1982.
- [DYH82] R. L. DeValois, E. W. Yund, and N. Hepler, "The orientation and direction selectivity of cells in macaque visual cortex," *Vision Res.*, vol. 22, no. 5, pp. 531-544, 1982.
- [DH87] P. Duhamel and H. H'Mida, "New 2ⁿ DCT algorithms suitable for VLSI implementation," *IEEE Int'l Conf. Acoustics, Speech, Signal Process.*, 1805-1808, April 1987.
- [Dau88] J. G. Daugman, "Complete discrete 2-D Gabor transforms by neural networks for image analysis and compression," *IEEE Trans. Acoustics, Speech, Signal Process.*, vol. 36, no. 7, pp. 1169-1179, July 1988.
- [DG90] P. Duhamel and C. Guillemot, "Polynomial transform computation of the 2-D DCT," *IEEE Int. Conf. Acoustic, Speech, Signal Process.*, pp. 1515-1518, April 1990.
- [Dal92] S. Daly, "Visible difference predictor: An algorithm for the assessment of image fidelity," *Proc. SPIE*, vol. 2, pp. 2-15, 1992.
- [Dal93] S. Daly, "The visible difference predictor: An algorithm for the assessment of image fidelity," *Digital Images and Human Vision*, pp. 179-206, 1993.

- [DA01] R. Dugad and N. Ahuja, "A fast scheme for image size change in the compressed domain," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 4, pp. 461–474, April 2001.
- [EF95] A. M. Eskicioglu and P. S. Fisher, "Image quality measures and their performance," *IEEE Trans. Commun.*, vol. 43, no. 12, pp. 2959-2965, December 1995.
- [EB98] M. P. Eckert and A. P. Bradley, "Perceptual quality metrics applied to still image compression," *Signal Process.*, vol. 70, no. 3, pp. 177-200, November 1998.
- [Fau79] O. D. Faugeras, "Digital color image processing within the framework of a human visual model," *IEEE Trans. Acoust. Speech Signal Process.*, vol. 27, no. 4, pp. 380-393, August 1979.
- [FW92] E. Feig and S. Winograd, "Fast algorithms for the discrete cosine transform", *IEEE Trans. Signal Process.*, vol. 40, no. 9, pp. 2174-2193, September 1992.
- [GRN78] N. Graham and J. G. Robson, J. Nachmias, "Grating summation in fovea and periphery," *Vision Res.*, vol. 18, no. 7, pp. 815-825, 1978.
- [Gir93] B. Girod, "What's wrong with mean-squared error," *Digital Images and Human Vision*, MIT Press, pp. 207-220, 1993.
- [Hec24] S. Hecht, "The visual discrimination of intensity and the Weber-Fechner law," *J. General Physiology*, vol. 7, no. 2, pp. 235-267, November 1924.
- [HW59] D. H. Hubel and T. N. Wiesel, "Receptive fields of single neurones in the cat's striate cortex," *J. Physiology*, vol. 148, no. 3, pp. 574-591, October 1959.

- [HW62] D. H. Hubel and T. N. Wiesel, "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex," *J. Physiology*, vol. 160, no. 1, pp. 106-154, January 1962.
- [HW68] D. H. Hubel and T. N. Wiesel, "Receptive fields and functional architecture of monkey striate cortex," *J. Physiology*, vol. 195, no. 1, pp. 215-243, March 1968.
- [HB85] M. T. Heideman and C. S. Burrus, "Multiply/add tradeoffs in length 2^n FFT algorithms," *IEEE Int'l Conf. Acoustics, Speech, Signal Process.*, 780-783, April 1985.
- [HH93] A. Hossen, and U. Heute, "Fully adaptive evaluation of sub-band DFT," *IEEE Int'l Symp. Circuits and Syst.*, pp. 655-658, May 1993.
- [HWH98] Y. M. Huang, J. L. Wu, and C. T. Hsu, "A refined fast 2-D discrete cosine transform algorithm with regular butterfly structure", *IEEE Trans. Consumer Electronics*, vol. 44, no. 2, pp. 376-383, May 1998.
- [HW99] Y. M. Huang and J. L. Wu, "A refined fast 2-D discrete cosine transform algorithm", *IEEE Trans. Signal Process.*, vol. 47, no. 3, pp. 904-907, March 1999.
- [HK00] I. Höntsch and L. J. Karam, "Locally adaptive perceptual image coding," *IEEE Trans. Image Process.*, vol. 9, no. 9, pp. 1472-1483, September 2000.
- [HK02] I. Höntsch and L. J. Karam, "Adaptive image coding with perceptual distortion control," *IEEE Trans. Image Process.*, vol. 11, no. 3, pp. 213-222, March 2002.

- [HH04] A. Hossen and U. Heute, "Two-dimensional subband transforms: theory and applications," *IEE Proc.-Vis. Image Signal Process.*, vol. 151, no. 5, pp. 389-399, October 2004.
- [ISO94] *Information Technology – Digital Compression and Coding of Continuous-Tone Still Images: Requirements and Guidelines*, ISO/IEC 10918-1, 1994.
- [ISO00] *Information Technology – JPEG 2000 Image Coding System - Part 1: Core Coding System*, ISO/IEC 15444-1, 2000.
- [Jai79] A. K. Jain, "A sinusoidal family of unitary transforms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-1, no. 4, pp. 356-365, October 1979.
- [Joh80] J. D. Johnston, "A filter family designed for use in quadrature mirror filter banks," *IEEE Int'l Conf. Acoustics, Speech, Signal Process.*, vol. 5, pp. 291-294, April 1980.
- [JJS93] N. Jayant, J. Johnston, and R. Safranek, "Signal compression based on models on human perception," *Proc. IEEE*, vol. 81, no. 10, pp. 1385-1422, October 1993.
- [JMM96] S. H. Jung, S. K. Mitra, and D. Mukherjee, "Subband DCT: definition, analysis, and applications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 3, pp. 273-286, June 1996.
- [JT01] J. Liang and T. D. Tran, "Fast multiplierless approximation of the DCT with the lifting scheme", *IEEE Trans. Signal Process.*, vol. 49, no. 12, pp. 3032-3044, December 2001.

- [JF02] J. Jiang and G. Feng, "The spatial relationship of DCT coefficients between a block and its sub-blocks," *IEEE Trans. Signal Process.*, vol. 50, no. 5, pp. 1160–1169, May 2002.
- [Kel85] D. H. Kelly, "Visual processing of moving stimuli," *J. Opt. Soc. Amer.*, vol. 2, no. 2, pp. 216–225, February 1985.
- [Leg78a] G. E. Legge, "Sustained and transient mechanisms in human vision: temporal and spatial properties," *Vision Res.*, vol. 18, no. 1, pp. 69–81, 1978.
- [Leg78b] G. E. Legge, "Space domain properties of a spatial frequency channel in human vision," *Vision Res.*, vol. 18, no. 8, pp. 959–969, 1978.
- [LF80] G. E. Legge and J. M. Foley, "Contrast masking in human vision," *J. Opt. Soc. Am.*, vol. 70, no. 12, pp. 1458–1471, December 1980.
- [LLM89] C. Loeffler, A. Ligtenberg, and G. S. Moschytz, "Practical fast 1-D DCT algorithms with 11 multiplications," *IEEE Int'l Conf. Acoustics, Speech, Signal Process.*, vol. 2, pp. 988–991, May 1989.
- [LB90] C. J. Lloyd and R. J. Beaton, "Design of a spatio-chromatic human vision model for evaluating full-color display systems," *Proc. SPIE*, vol. 1249, no. 23, pp. 23–37, February 1990.
- [LB92] F. X. J. Lukas and Z. L. Budrikis, "Picture quality prediction based on a visual model," *IEEE Trans. Comm.*, vol. 30, no. 7, pp. 1679–1692, July 1982.
- [Lub93] J. Lubin, "The use of psychophysical data and models in the analysis of display system performance," *Digital Image and Human Vision*, pp. 163–178, 1993.

- [Lub95] J. Lubin, "A visual discrimination mode for image system design and evaluation," *Visual Models for Target Detection and Recognition*, pp. 207-220, 1995.
- [LMK98] B. Li, G. W. Meyer, and R. V. Klassen, "A comparison of two image quality models," *SPIE Conf. Human Vision, Electronic Imaging III*, vol. 3299, pp. 98-109, January 1998.
- [LK00] Y. K. Lai and C. C. J. Kuo, "A Haar wavelet approach to compressed image quality measurement," *J. Vis. Commun. Image Representation*, vol. 11, pp. 17-40, March 2000.
- [LDX03] W. S. Lin, L. Dong, and P. Xue, "Discriminative analysis of pixel difference towards picture quality prediction," *Proc. IEEE Int'l Conf. Image Processing*, vol. 3, pp. 193-196, September 2003.
- [LL07] Y. R. Lee and C. W. Lin, "Visual quality enhancement in DCT-domain spatial downscaling transcoding using generalized DCT decimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 8, pp. 1079–1084, August 2007.
- [LLP10] A.M. Liu, W.S. Lin, M. Paul, C.W. Deng, and F. Zhang, "Just noticeable difference for images with decomposition model for separating edge and textured regions," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 11, pp. 1648-1652, November 2010.
- [LK11] W. S. Lin and C. C. J. Kuo, "Perceptual visual quality metrics: a survey," *J. Vis. Commun. Image Representation*, vol. 22, no. 4, pp. 297-312, May 2011.

- [MS74] J. L. Mannos and D. J. Sakrison, "The effects of a visual fidelity criterion on the encoding of images," *IEEE Trans. Inform. Theory*, vol. 20, no. 4, pp. 525-536, July 1974.
- [MTT78] J. A. Movshon, I. D. Thompson, and D. J. Tolhurst, "Spatial and temporal contrast sensitivity of neurones in areas 17 and 18 of the cat's visual cortex," *J. Physiology*, vol. 283, pp. 101-120, October 1978.
- [Mak80] J. Makhoul, "A fast cosine transform in one and two dimensions," *IEEE Trans. Acoustics, Speech, Signal Process.*, vol. 28, no. 1, pp. 27-34, February 1980.
- [MSP90] S. K. Mitra, O. V. Shentov, and M. R. Petraglia, "A method for fast approximate computation of discrete time transform," *IEEE Int. Conf. Acoustic, Speech, Signal Process.*, pp. 2025-2028, April 1990.
- [Mit01] S. K. Mitra, *Digital Signal Processing: A Computer Based Approach*. (New York: McGraw-Hill, 2001).
- [MM05] J. Mukherjee and S. K. Mitra, "Arbitrary resizing of images in DCT space," *IEE Proc.-Vis. Image Signal Process.*, vol. 152, no. 2, pp. 155-164, April 2005.
- [MSO07] R. Muralishankar, A. Sangwan, and D. O'Shaughnessy, "Theoretical complex cepstrum of DCT and warped DCT filters," *IEEE Signal Process. Letters*, vol. 14, no. 5, pp. 367-370, May 2007.
- [NB67] F. L. van Nes and M. A. Bouman, "Spatial modulation transfer in the human eye," *J. Opt. Soc. Amer.*, vol. 57, no. 3, pp. 401-406, 1967.
- [Nil85] N. B. Nill, "A visual model weighted cosine transform for image compression and quality assessment," *IEEE Trans. Commun.*, vol. 33, no. 6, pp. 551-557, June 1985.

- [NLS86] K. N. Ngan, K. S. Leong, and H. Singh, "Cosine transform coding incorporating human visual system model," *SPIE Fiber*, pp. 165-171, September 1986.
- [NAW93] R. Navarro, P. Artal, and D. R. Williams, "Modulation transfer of the human eye as a function of retinal eccentricity," *J. Opt. Soc. Amer.*, vol. 10, no. 2, pp. 201-212, February 1993.
- [Ost85] G. A. Osterberg, "Topography of the layer of rods and cones in the human retina," *Acta Ophthalmologica*, vol. 6, pp. 1-102, 1935.
- [PDT77] G. F. Poggio, R. W. Doty, and W. H. Talbot, "Foveal striate cortex of behaving monkey: single-neuron responses to square-wave gratings fixation of gaze," *J. Neurophysiology*, vol. 40, no. 6, pp. 1369-1391, November 1977.
- [PW84] G. C. Philips and H. R. Wilson, "Orientation bandwidths of spatial mechanisms measured by masking," *J. Opt. Soc. Amer.*, vol. 1, no. 2, pp. 226-232, February 1984.
- [PMP91] H. A. Peterson, H. Peng, J. H. Morgan, and W. B. Pennebaker, "Quantization of color image components in the DCT domain," *Human Vision, Visual Process. Digital Display II*, Proc. SPIE, vol. 1453, pp. 210-222, 1991.
- [Pet92] H. A. Peterson, "DCT basis function visibility threshold in RGB space", in J. Morreale, ed., *SID International Symposium Digest of Technical Papers*, Society for Information Display, pp. 677-680, 1992.
- [PAW93a] H. A. Peterson, A. J. Ahumada, and A. B. Watson, "An improved detection model for DCT coefficient quantization," *Proc. SPIE*, vol. 1913, pp. 191-201, 1992.

- [PAW93b] H. A. Peterson, A. J. Ahumada, and A. B. Watson, "The visibility of DCT quantization noise," *Society for Information Display Digest of Technical Papers*, pp. 942-945, 1993.
- [PJJ94] J. Park, J. M. Jo, and J. Jeong, "Some adaptive quantizers for HDTV image compression," *Signal Process. HDTV*, 1994.
- [PD96] J. Prado and P. Duhamel, "A polynomial transform-based computation of the 2-D DCT with minimum multiplicative complexity", *IEEE Int. Conf. Acoustic, Speech, Signal Process.*, pp. 1347-1350, May 1996.
- [PPO03] H. W. Park, Y. S. Park, and S. K. Oh, "L/M-fold image resizing in block-DCT domain using symmetric convolution," *IEEE Trans. Image Process.*, vol. 12, no. 9, pp. 1016-1034, September 2003.
- [PP06] Y. S. Park and H. W. Park, "Arbitrary-ratio image resizing using fast DCT of composite length for DCT-based transcoder," *IEEE Trans. Image Process.*, vol. 15, no. 2, pp. 494-500, February 2006.
- [Rob66] J. G. Robson, "Spatial and temporal contrast sensitivity functions of the visual system," *J. Opt. Soc. Amer.*, vol. 56, no. 8, pp. 1141-1142, February 1966.
- [RD70] W. D. Ray and R. M. Driver, "Further decomposition of the Karhunen-Loeve series representation of a stationary random process," *IEEE Trans. Inform. Theory*, vol. 16, no. 6, pp. 663-668, November 1970.
- [RG81] J. G. Robson and N. Graham, "Probability summation and regional variation in contrast sensitivity across the visual field," *Vision Res.*, vol. 21, no. 3, pp. 409-418, 1981.

- [RAW97] A. M. Rohaly, A. J. Ahumada, and A. B. Watson, "Object detection in natural backgrounds predicted by discrimination performance and models," *Vision Res.*, vol. 37, no. 23, pp. 3225-3235, December 1997.
- [RKH10] K. R. Rao, D. N. Kim, and J. J. Hwang, *Fast Fourier Transform - Algorithms and Applications*. (New York: Springer, 2010).
- [Sch56] O. H. Schade, "Optical and photoelectric analog of the eye," *J. Opt. Soc. Am.*, vol. 46, no. 9, pp. 721-739, September 1956.
- [SJ72] C. F. Stromeyer and B. Julesz, "Spatial-frequency masking in vision: Critical bands and spread of masking," *J. Opt. Soc. Am.*, vol. 62, no. 10, pp. 1221-1232, October 1972.
- [Sto72] T. G. Stockham, "Image processing in the context of a visual model," *Proc. IEEE*, vol. 60, no. 7, pp. 828-842, July 1972.
- [SYD87] M. I. Sezan, K. L. Yip, and S. Daly, "Uniform perceptual quantization: Applications to digital radiography," *IEEE Trans. Sys. Man Cybernetics*, vol. 17, no. 4, pp. 622-634, 1987.
- [SJ89] R. J. Safranek and J. D. Johnston, "A perceptually tuned subband image coder with image dependent quantization and post-quantization," *IEEE Int. Conf. Acoustic, Speech, Signal Process.*, pp. 1945-1948, May 1989.
- [SE00] D. Santa-Cruz and T. Ebrahimi, "A study of JPEG 2000 still image coding versus other standards," *Proc. X European Signal Processing Conf.*, vol. 2, pp. 673-676, September 2000.
- [ST04] C. L. Salazar and T. D. Tran "On resizing images in the DCT domain," *IEEE Int'l Conf. Image Process.*, vol. 4, pp. 2797-2800, October 2004.

- [SC07] H. Shu and L. P. Chau, "A resizing algorithm with two-stage realization for DCT-based transcoding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 2, pp. 248–253, February 2007.
- [TH94a] P. C. Teo and D. J. Heeger, "Perceptual image distortion," *Proc. SPIE*, vol. 2179, pp. 127-141, 1994.
- [TH94b] P. C. Teo and D. J. Heeger, "Perceptual image distortion," *Proc. IEEE Int. Conf. Image Process.*, vol. 2, pp. 982-986, November 1994.
- [TS96] T. D. Tran and R. Safranek, "A locally perceptual masking threshold model for image coding," *IEEE Int. Conf. Acoustic, Speech, Signal Process.*, pp. 1882-1885, May 1996.
- [TV98] H. Y. Tong and A. N. Venetsanopoulos, "A perceptual model for JPEG applications based on block classification, texture masking, and luminance masking," *Proc. IEEE Int'l Conf. Image Process.*, 1998.
- [URL01] http://www.siliconimaging.com/hdtv_images.htm, accessed December 2007.
- [VN84] M. Vetterli and H. J. Nussbaumer, "Simple FFT and DCT algorithms with reduced number of operations," *Signal Process.*, vol. 6, no. 4, pp. 267–278, August 1984.
- [VN86] M. Vetterli and A. Ligtenberg, "A discrete Fourier-cosine transform chip," *IEEE J. of Selected Areas in Commun.*, vol. 4, no. 1, pp. 49-61, January 1986.
- [VKR88] S. Venkataraman, V. R. Kanchan, K. R. Rao, and M. Mohanty, "Discrete transforms via the Walsh–Hadamard transform", *Signal Process.*, vol. 14, no. 4, pp. 371-382, June 1988.

- [VS98] A. Vetro, and H. Sun, "On the motion compensation within a down-conversion decoder," *SPIE Journal of Electronic Imaging*, vol. 7, no. 3, pp. 616-627, July 1998.
- [VCS03] A. Vetro, C. Christopoulos, and H. Sun, "Video transcoding architectures and techniques: An overview," *IEEE Signal Process. Mag.*, vol. 20, no. 2, pp. 18-29, March 2003.
- [VQE03] VQEG. (2003). *Final report from video quality experts group on the validation of objective models of video quality assessment*, Available: ftp://vqeg.its.bldrdoc.gov/Documents/VQEG_Approved_Final_Reports/VQEGII_Final_Report.pdf
- [Wat79] A. B. Watson, "Probability summation over time," *Vision Res.*, vol. 19, no. 5, pp. 515-522, 1979.
- [Wat82] A. B. Watson, "Summation of grating patches indicates many types of detectors at one retinal location," *Vision Res.*, vol. 22, no. 1, pp. 17-25, 1982.
- [WR84] H. R. Wilson and D. Regan, "Spatial frequency adaptation and grating discrimination: Prediction of a line element model," *J. Opt. Soc. Am.*, vol. 1, no. 11, pp. 1091-1096, 1984.
- [WD85] M. A. Webster and R. L. DeValois, "Relationship between spatial-frequency and orientation tuning of striate-cortex cells," *J. Opt. Soc. Amer.*, vol. 2, no. 7, pp. 1124-1132, 1985.
- [Wes86] G. Westheimer, "The eye as an optical instrument," in *Handbook of Human Perception and Performance*, Wiley and Sons, 1986.

- [Wat87] A. B. Watson, "The cortex transform: Rapid computation of simulated neural images," *Computer Vision, Graphics, and Image Process.*, vol. 39, no. 3, pp. 311-327, 1987.
- [WP89a] H. R. Wu and F. J. Paoloni, "A two-dimensional fast cosine transform algorithm", *Proc. IEEE Int'l Conf. Image Process.*, pp. 50–54, September 1989.
- [WP89b] H. R. Wu and F. J. Paoloni, "A structural approach to two-dimensional direct fast discrete cosine transform algorithms", *Proc. Int'l Symp. Computer Architectures and DSP*, October 1989.
- [WLM90] H. R. Wilson, D. Levi, L. Maffei, J. Rovamo, and R. DeValois, "The perception of form: Retina to striate cortex," *Visual Perception: The Neurophysiological Foundations*, in L. Spillman and J. Werner Eds., pp. 231-272, Academic Press, 1990.
- [WP91] H. R. Wu and F. J. Paoloni, "A two-dimensional fast cosine transform algorithm based on Hou's approach", *IEEE Trans. Signal Process.*, vol. 39, no. 2, pp. 544-546, February 1991.
- [Wal92] G. K. Wallace, "The JPEG still picture compression standard," *IEEE Trans. Consumer Electronics*, vol. 38, no. 1, pp. xviii - xxxiv, February 1992.
- [Wat93] A. B. Watson, "DCT quantization matrices visually optimized for individual images," *Proc. SPIE Human Vision, Visual Proc., Digital Display IV*, Feb. 1993, pp. 202–216.
- [WHM97] A. B. Watson, G. Y. Yang, J. A. Solomon, and J. Villasenor, "Visibility of wavelet quantization noise," *IEEE Trans. Image Process.*, vol. 6, no. 8, pp. 1164-1175, August 1997.

- [WPD99] R. B. Wolfgang, C. I. Podilchuk, and E. J. Delp, "Perceptual watermarks for digital images and video," *Proc. IEEE*, vol. 87, no. 7, pp. 1108-1126, July 1999.
- [WSS00] M. J. Weinberger, G. Seroussi, and G. Sapiro, "The LOCO-I lossless image compression algorithm: Principles and standardization into JPEG-LS," *IEEE Trans. Image Process.*, vol. 9, no. 8, pp. 1309-1324, August 2000.
- [WB02] Z. Wang and A. C. Bovik, "A universal image quality index," *IEEE Signal Proc. Lett.*, vol. 9, no. 3, pp. 81-84, March 2002.
- [WBS04] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600-612, April 2004.
- [WA05] A. B. Watson and A. J. Ahumada, "A standard model for foveal detection of spatial contrast," *J. Vision*, vol. 5, no. 9, pp. 717-740, October 2005.
- [WN09] Z.Y. Wei and K.N. Ngan, "Spatio-temporal just noticeable distortion profile for grey scale image/video in DCT domain," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 3, pp. 337-346, March 2009.
- [YLL03] X. K. Yang, W. S. Lin, Z. K. Lu, E. P. Ong, and S. S. Yao, "On incorporating just-noticeable-distortion profile into motion-compensated prediction for video compression," *IEEE Int'l Conf. Image Process.*, vol. 3, pp. 833-836, September 2003.
- [YGO05] W. Yin, D. Goldfarb, and S. Osher, "Image cartoon-texture decomposition and feature selection using the total variation

- regularized L1 functional," *Variational, Geometric, and Level Set Methods in Computer Vision*, pp. 73-84, 2005.
- [YLL05] X. K. Yang, W. S. Lin, Z. K. Lu, E. P. Ong, and S. S. Yao, "Just noticeable distortion model and its applications in video coding," *Signal Process.: Image Commun.*, vol. 20, no. 7, pp. 662-680, August 2005.
- [ZLX05] X. H. Zhang, W. S. Lin, and P. Xue, "Improved estimation for just-noticeable visual distortion," *Signal Process.*, vol. 85, no. 4, pp. 795-808, April 2005.
- [ZB06] Z. Wang and A. C. Bovik, *Modern image quality assessment*, Morgan and Claypool, 2006.
- [ZLX08] X. H. Zhang, W. S. Lin, and P. Xue, "Just-noticeable-difference estimation with pixels in images," *J. Vis. Commun. Image Representation*, vol. 19, no. 1, pp. 30-41, January 2008.

Appendix A

Derivation of Aliased Components in Subband Discrete Cosine Transform

The aliased components of the four transforms in SBDCT are obtained as follows:

$$\begin{aligned}
& \frac{C(0, \mathbf{k}, \mathbf{n})}{\alpha_{k_1} \alpha_{k_2} W_c^{k_1/2N} W_c^{k_2/2N}} \\
&= 4 \sum_{p_1=0}^{N'-1} \sum_{p_2=0}^{N'-1} c(0, \mathbf{p}, \mathbf{n}) W_c^{(2p_1+1)k_1/N} W_c^{(2p_2+1)k_2/N} \\
&= W_c^{k_1/2N} W_c^{k_2/2N} \sum_{p_1=0}^{N'-1} \sum_{p_2=0}^{N'-1} \left[c(2p_1, 2p_2, \mathbf{n}) W_c^{(4p_1+1)k_1/2N} W_c^{(4p_2+1)k_2/2N} \right. \\
&\quad + c(2p_1+1, 2p_2, \mathbf{n}) W_c^{(4p_1+3)k_1/2N} W_c^{(4p_2+1)k_2/2N} + c(2p_1, 2p_2+1, \mathbf{n}) W_c^{(4p_1+1)k_1/2N} W_c^{(4p_2+3)k_2/2N} \\
&\quad \left. + c(2p_1+1, 2p_2+1, \mathbf{n}) W_c^{(4p_1+3)k_1/2N} W_c^{(4p_2+3)k_2/2N} \right] \\
&+ W_c^{k_1/2N} W_s^{k_2/2N} \sum_{p_1=0}^{N'-1} \sum_{p_2=0}^{N'-1} \left[-c(2p_1, 2p_2, \mathbf{n}) W_c^{(4p_1+1)k_1/2N} W_s^{(4p_2+1)k_2/2N} \right. \\
&\quad - c(2p_1+1, 2p_2, \mathbf{n}) W_c^{(4p_1+3)k_1/2N} W_s^{(4p_2+1)k_2/2N} + c(2p_1, 2p_2+1, \mathbf{n}) W_c^{(4p_1+1)k_1/2N} W_s^{(4p_2+3)k_2/2N} \\
&\quad \left. + c(2p_1+1, 2p_2+1, \mathbf{n}) W_c^{(4p_1+3)k_1/2N} W_s^{(4p_2+3)k_2/2N} \right] \\
&+ W_s^{k_1/2N} W_c^{k_2/2N} \sum_{p_1=0}^{N'-1} \sum_{p_2=0}^{N'-1} \left[-c(2p_1, 2p_2, \mathbf{n}) W_s^{(4p_1+1)k_1/2N} W_c^{(4p_2+1)k_2/2N} \right. \\
&\quad + c(2p_1+1, 2p_2, \mathbf{n}) W_s^{(4p_1+3)k_1/2N} W_c^{(4p_2+1)k_2/2N} - c(2p_1, 2p_2+1, \mathbf{n}) W_s^{(4p_1+1)k_1/2N} W_c^{(4p_2+3)k_2/2N} \\
&\quad \left. + c(2p_1+1, 2p_2+1, \mathbf{n}) W_s^{(4p_1+3)k_1/2N} W_c^{(4p_2+3)k_2/2N} \right] \\
&+ W_s^{k_1/2N} W_s^{k_2/2N} \sum_{p_1=0}^{N'-1} \sum_{p_2=0}^{N'-1} \left[c(2p_1, 2p_2, \mathbf{n}) W_s^{(4p_1+1)k_1/2N} W_s^{(4p_2+1)k_2/2N} \right. \\
&\quad - c(2p_1+1, 2p_2, \mathbf{n}) W_s^{(4p_1+3)k_1/2N} W_s^{(4p_2+1)k_2/2N} - c(2p_1, 2p_2+1, \mathbf{n}) W_s^{(4p_1+1)k_1/2N} W_s^{(4p_2+3)k_2/2N} \\
&\quad \left. + c(2p_1+1, 2p_2+1, \mathbf{n}) W_s^{(4p_1+3)k_1/2N} W_s^{(4p_2+3)k_2/2N} \right].
\end{aligned} \tag{A.1}$$

Since the cosine basis functions can be rewritten as

$$\begin{aligned}
W_c^{(4n+1)(k-N)\pi/2N} &= \cos\left(\frac{4n+1}{2N}k\pi - \frac{4n+1}{2N}N\pi\right) \\
&= \cos\left(\frac{4n+1}{2N}k\pi\right)\cos\left(\frac{4n+1}{2}\pi\right) + \sin\left(\frac{4n+1}{2N}k\pi\right)\sin\left(\frac{4n+1}{2}\pi\right) \tag{A.2} \\
&= \sin\left(\frac{4n+1}{2N}k\pi\right),
\end{aligned}$$

and

$$\begin{aligned}
W_c^{(4n+3)(k-N)\pi/2N} &= \cos\left(\frac{4n+3}{2N}k\pi - \frac{4n+3}{2N}N\pi\right) \\
&= \cos\left(\frac{4n+3}{2N}k\pi\right)\cos\left(\frac{4n+3}{2}\pi\right) + \sin\left(\frac{4n+3}{2N}k\pi\right)\sin\left(\frac{4n+3}{2}\pi\right) \\
&= -\sin\left(\frac{4n+3}{2N}k\pi\right).
\end{aligned} \tag{A.3}$$

Therefore, we have

$$\begin{aligned}
&\frac{C(0, \mathbf{k}, \mathbf{n})}{\alpha_{k_1} \alpha_{k_2} W_c^{k_1/2N} W_c^{k_2/2N}} \\
&= \alpha_{k_1} \alpha_{k_2} W_c^{k_1/2N} W_c^{k_2/2N} \left[W_c^{k_1/2N} W_c^{k_2/2N} C(\mathbf{k}, \mathbf{n}) - W_c^{k_1/2N} W_s^{k_2/2N} C(k_1, k'_2, \mathbf{n}) \right. \\
&\quad \left. - W_s^{k_1/2N} W_c^{k_2/2N} C(k'_1, k_2, \mathbf{n}) + W_s^{k_1/2N} W_s^{k_2/2N} C(k'_1, k'_2, \mathbf{n}) \right].
\end{aligned} \tag{A.4}$$

The second transform $C(1, \mathbf{k}, \mathbf{n})$ is rewritten as

$$\begin{aligned}
&\frac{C(1, \mathbf{k}, \mathbf{n})}{\alpha_{k_1} \alpha_{k_2} W_c^{k_1/2N} W_s^{k_2/2N}} \\
&= 4 \sum_{p_1=0}^{N'-1} \sum_{p_2=0}^{N'-1} c(1, \mathbf{p}, \mathbf{n}) W_c^{(2p_1+1)k_1/N} W_s^{(2p_2+1)k_2/N} \\
&= W_c^{k_1/2N} W_s^{k_2/2N} \sum_{p_1=0}^{N'-1} \sum_{p_2=0}^{N'-1} \left[c(2p_1, 2p_2, \mathbf{n}) W_c^{(4p_1+1)k_1/2N} W_c^{(4p_2+1)k_2/2N} \right. \\
&\quad + c(2p_1+1, 2p_2, \mathbf{n}) W_c^{(4p_1+3)k_1/2N} W_c^{(4p_2+1)k_2/2N} + c(2p_1, 2p_2+1, \mathbf{n}) W_c^{(4p_1+1)k_1/2N} W_c^{(4p_2+3)k_2/2N} \\
&\quad \left. + c(2p_1+1, 2p_2+1, \mathbf{n}) W_c^{(4p_1+3)k_1/2N} W_c^{(4p_2+3)k_2/2N} \right] \\
&+ W_s^{k_1/2N} W_s^{k_2/2N} \sum_{p_1=0}^{N'-1} \sum_{p_2=0}^{N'-1} \left[-c(2p_1, 2p_2, \mathbf{n}) W_s^{(4p_1+1)k_1/2N} W_c^{(4p_2+1)k_2/2N} \right. \\
&\quad + c(2p_1+1, 2p_2, \mathbf{n}) W_s^{(4p_1+3)k_1/2N} W_c^{(4p_2+1)k_2/2N} - c(2p_1, 2p_2+1, \mathbf{n}) W_s^{(4p_1+1)k_1/2N} W_c^{(4p_2+3)k_2/2N} \\
&\quad \left. + c(2p_1+1, 2p_2+1, \mathbf{n}) W_s^{(4p_1+3)k_1/2N} W_c^{(4p_2+3)k_2/2N} \right] \\
&+ W_c^{k_1/2N} W_c^{k_2/2N} \sum_{p_1=0}^{N'-1} \sum_{p_2=0}^{N'-1} \left[c(2p_1, 2p_2, \mathbf{n}) W_c^{(4p_1+1)k_1/2N} W_s^{(4p_2+1)k_2/2N} \right. \\
&\quad + c(2p_1+1, 2p_2, \mathbf{n}) W_c^{(4p_1+3)k_1/2N} W_s^{(4p_2+1)k_2/2N} - c(2p_1, 2p_2+1, \mathbf{n}) W_c^{(4p_1+1)k_1/2N} W_s^{(4p_2+3)k_2/2N} \\
&\quad \left. - c(2p_1+1, 2p_2+1, \mathbf{n}) W_c^{(4p_1+3)k_1/2N} W_s^{(4p_2+3)k_2/2N} \right] \\
&+ W_s^{k_1/2N} W_c^{k_2/2N} \sum_{p_1=0}^{N'-1} \sum_{p_2=0}^{N'-1} \left[-c(2p_1, 2p_2, \mathbf{n}) W_s^{(4p_1+1)k_1/2N} W_s^{(4p_2+1)k_2/2N} \right. \\
&\quad + c(2p_1+1, 2p_2, \mathbf{n}) W_s^{(4p_1+3)k_1/2N} W_s^{(4p_2+1)k_2/2N} + c(2p_1, 2p_2+1, \mathbf{n}) W_s^{(4p_1+1)k_1/2N} W_s^{(4p_2+3)k_2/2N} \\
&\quad \left. - c(2p_1+1, 2p_2+1, \mathbf{n}) W_s^{(4p_1+3)k_1/2N} W_s^{(4p_2+3)k_2/2N} \right]
\end{aligned} \tag{A.5}$$

$$\frac{C(1, \mathbf{k}, \mathbf{n})}{\alpha_{k_1} \alpha_{k_2} W_c^{k_1/2N} W_s^{k_2/2N}} = W_c^{k_1/2N} W_s^{k_2/2N} \left[C(\mathbf{k}, \mathbf{n}) + \frac{W_c^{k_2/2N}}{W_s^{k_2/2N}} C(k_1, k'_2, \mathbf{n}) - \frac{W_s^{k_1/2N}}{W_c^{k_1/2N}} C(k'_1, k_2, \mathbf{n}) - \frac{W_s^{k_1/2N} W_c^{k_2/2N}}{W_c^{k_1/2N} W_s^{k_2/2N}} C(\mathbf{k}', \mathbf{n}) \right].$$

The aliased components of the third transform $C(2, \mathbf{k}, \mathbf{n})$ are

$$\begin{aligned} & \frac{C(2, \mathbf{k}, \mathbf{n})}{\alpha_{k_1} \alpha_{k_2} W_s^{k_1/2N} W_c^{k_2/2N}} \\ &= 4 \sum_{p_1=0}^{N'-1} \sum_{p_2=0}^{N'-1} c(2, \mathbf{p}, \mathbf{n}) W_s^{(2p_1+1)k_1/2N} W_c^{(2p_2+1)k_2/2N} \\ &= W_s^{k_1/2N} W_c^{k_2/2N} \sum_{p_1=0}^{N'-1} \sum_{p_2=0}^{N'-1} \left[c(2p_1, 2p_2, \mathbf{n}) W_c^{(4p_1+1)k_1/2N} W_c^{(4p_2+1)k_2/2N} \right. \\ & \quad + c(2p_1+1, 2p_2, \mathbf{n}) W_c^{(4p_1+3)k_1/2N} W_c^{(4p_2+1)k_2/2N} + c(2p_1, 2p_2+1, \mathbf{n}) W_c^{(4p_1+1)k_1/2N} W_c^{(4p_2+3)k_2/2N} \\ & \quad \left. + c(2p_1+1, 2p_2+1, \mathbf{n}) W_c^{(4p_1+3)k_1/2N} W_c^{(4p_2+3)k_2/2N} \right] \\ &+ W_c^{k_1/2N} W_c^{k_2/2N} \sum_{p_1=0}^{N'-1} \sum_{p_2=0}^{N'-1} \left[c(2p_1, 2p_2, \mathbf{n}) W_s^{(4p_1+1)k_1/2N} W_c^{(4p_2+1)k_2/2N} \right. \\ & \quad - c(2p_1+1, 2p_2, \mathbf{n}) W_s^{(4p_1+3)k_1/2N} W_c^{(4p_2+1)k_2/2N} + c(2p_1, 2p_2+1, \mathbf{n}) W_s^{(4p_1+1)k_1/2N} W_c^{(4p_2+3)k_2/2N} \\ & \quad \left. - c(2p_1+1, 2p_2+1, \mathbf{n}) W_s^{(4p_1+3)k_1/2N} W_c^{(4p_2+3)k_2/2N} \right] \\ &+ W_s^{k_1/2N} W_s^{k_2/2N} \sum_{p_1=0}^{N'-1} \sum_{p_2=0}^{N'-1} \left[-c(2p_1, 2p_2, \mathbf{n}) W_c^{(4p_1+1)k_1/2N} W_s^{(4p_2+1)k_2/2N} \right. \\ & \quad - c(2p_1+1, 2p_2, \mathbf{n}) W_c^{(4p_1+3)k_1/2N} W_s^{(4p_2+1)k_2/2N} + c(2p_1, 2p_2+1, \mathbf{n}) W_c^{(4p_1+1)k_1/2N} W_s^{(4p_2+3)k_2/2N} \\ & \quad \left. + c(2p_1+1, 2p_2+1, \mathbf{n}) W_c^{(4p_1+3)k_1/2N} W_s^{(4p_2+3)k_2/2N} \right] \\ &+ W_c^{k_1/2N} W_s^{k_2/2N} \sum_{p_1=0}^{N'-1} \sum_{p_2=0}^{N'-1} \left[-c(2p_1, 2p_2, \mathbf{n}) W_s^{(4p_1+1)k_1/2N} W_s^{(4p_2+1)k_2/2N} \right. \\ & \quad + c(2p_1+1, 2p_2, \mathbf{n}) W_s^{(4p_1+3)k_1/2N} W_s^{(4p_2+1)k_2/2N} + c(2p_1, 2p_2+1, \mathbf{n}) W_s^{(4p_1+1)k_1/2N} W_s^{(4p_2+3)k_2/2N} \\ & \quad \left. - c(2p_1+1, 2p_2+1, \mathbf{n}) W_s^{(4p_1+3)k_1/2N} W_s^{(4p_2+3)k_2/2N} \right] \\ &= W_s^{k_1/2N} W_c^{k_2/2N} \left[C(\mathbf{k}, \mathbf{n}) - \frac{S_{2N}^{k_2}}{C_{2N}^{k_2}} C(k_1, k'_2, \mathbf{n}) + \frac{C_{2N}^{k_1}}{S_{2N}^{k_1}} C(k'_1, k_2, \mathbf{n}) - \frac{C_{2N}^{k_1} S_{2N}^{k_2}}{S_{2N}^{k_1} C_{2N}^{k_2}} C(\mathbf{k}', \mathbf{n}) \right]. \end{aligned} \tag{A.6}$$

The aliased components of the third transform $C(3, \mathbf{k}, \mathbf{n})$ are

$$\begin{aligned} & \frac{C(3, \mathbf{k}, \mathbf{n})}{\alpha_{k_1} \alpha_{k_2} W_s^{k_1/2N} W_s^{k_2/2N}} \\ &= 4 \sum_{p_1=0}^{N'-1} \sum_{p_2=0}^{N'-1} c(3, \mathbf{p}, \mathbf{n}) W_s^{(2p_1+1)k_1/N} W_s^{(2p_2+1)k_2/N} \end{aligned}$$

$$\begin{aligned}
&= W_s^{k_1/2N} W_s^{k_2/2N} \sum_{p_1=0}^{N'-1} \sum_{p_2=0}^{N'-1} \left[c(2p_1, 2p_2, \mathbf{n}) W_c^{(4p_1+1)k_1/2N} W_c^{(4p_2+1)k_2/2N} \right. \\
&\quad + c(2p_1+1, 2p_2, \mathbf{n}) W_c^{(4p_1+3)k_1/2N} W_c^{(4p_2+1)k_2/2N} + c(2p_1, 2p_2+1, \mathbf{n}) W_c^{(4p_1+1)k_1/2N} W_c^{(4p_2+3)k_2/2N} \\
&\quad \left. + c(2p_1+1, 2p_2+1, \mathbf{n}) W_c^{(4p_1+3)k_1/2N} W_c^{(4p_2+3)k_2/2N} \right] \\
&+ W_c^{k_1/2N} W_s^{k_2/2N} \sum_{p_1=0}^{N'-1} \sum_{p_2=0}^{N'-1} \left[c(2p_1, 2p_2, \mathbf{n}) W_s^{(4p_1+1)k_1/2N} W_c^{(4p_2+1)k_2/2N} \right. \\
&\quad - c(2p_1+1, 2p_2, \mathbf{n}) W_s^{(4p_1+3)k_1/2N} W_c^{(4p_2+1)k_2/2N} + c(2p_1, 2p_2+1, \mathbf{n}) W_s^{(4p_1+1)k_1/2N} W_c^{(4p_2+3)k_2/2N} \\
&\quad \left. - c(2p_1+1, 2p_2+1, \mathbf{n}) W_s^{(4p_1+3)k_1/2N} W_c^{(4p_2+3)k_2/2N} \right] \\
&+ W_s^{k_1/2N} W_c^{k_2/2N} \sum_{p_1=0}^{N'-1} \sum_{p_2=0}^{N'-1} \left[c(2p_1, 2p_2, \mathbf{n}) W_c^{(4p_1+1)k_1/2N} W_s^{(4p_2+1)k_2/2N} \right. \\
&\quad + c(2p_1+1, 2p_2, \mathbf{n}) W_c^{(4p_1+3)k_1/2N} W_s^{(4p_2+1)k_2/2N} - c(2p_1, 2p_2+1, \mathbf{n}) W_c^{(4p_1+1)k_1/2N} W_s^{(4p_2+3)k_2/2N} \\
&\quad \left. - c(2p_1+1, 2p_2+1, \mathbf{n}) W_c^{(4p_1+3)k_1/2N} W_s^{(4p_2+3)k_2/2N} \right] \\
&+ W_c^{k_1/2N} W_c^{k_2/2N} \sum_{p_1=0}^{N'-1} \sum_{p_2=0}^{N'-1} \left[c(2p_1, 2p_2, \mathbf{n}) W_s^{(4p_1+1)k_1/2N} W_s^{(4p_2+1)k_2/2N} \right. \\
&\quad - c(2p_1+1, 2p_2, \mathbf{n}) W_s^{(4p_1+3)k_1/2N} W_s^{(4p_2+1)k_2/2N} - c(2p_1, 2p_2+1, \mathbf{n}) W_s^{(4p_1+1)k_1/2N} W_s^{(4p_2+3)k_2/2N} \\
&\quad \left. + c(2p_1+1, 2p_2+1, \mathbf{n}) W_s^{(4p_1+3)k_1/2N} W_s^{(4p_2+3)k_2/2N} \right] \\
&= W_s^{k_1/2N} W_s^{k_2/2N} \left[C(\mathbf{k}, \mathbf{n}) + \frac{W_c^{k_2/2N}}{W_s^{k_2/2N}} C(k_1, k'_2, \mathbf{n}) + \frac{W_c^{k_1/2N}}{W_s^{k_1/2N}} C(k'_1, k_2, \mathbf{n}) + \frac{W_c^{k_1/2N} W_c^{k_2/2N}}{W_s^{k_1/2N} W_s^{k_2/2N}} C(\mathbf{k}', \mathbf{n}) \right]
\end{aligned} \tag{A.7}$$

To summarize, we have the aliased components for the four transforms in SBDCT as

$$\begin{aligned}
&4\alpha_{k_1} \alpha_{k_2} W_c^{k_1/2N} W_c^{k_2/2N} C(0, \mathbf{k}, \mathbf{n}) \\
&= \alpha_{k_1} \alpha_{k_2} \left(W_c^{k_1/2N} \right)^2 \left(W_c^{k_2/2N} \right)^2 \left[C(\mathbf{k}, \mathbf{n}) - \frac{W_s^{k_2/2N}}{W_c^{k_2/2N}} C(k_1, k'_2, \mathbf{n}) \right. \\
&\quad \left. - \frac{W_s^{k_1/2N}}{W_c^{k_1/2N}} C(k'_1, k_2, \mathbf{n}) + \frac{W_s^{k_1/2N} W_s^{k_2/2N}}{W_c^{k_1/2N} W_c^{k_2/2N}} C(\mathbf{k}', \mathbf{n}) \right],
\end{aligned} \tag{A.8}$$

$$\begin{aligned}
&4\alpha_{k_1} \alpha_{k_2} W_c^{k_1/2N} W_s^{k_2/2N} C(1, \mathbf{k}, \mathbf{n}) \\
&= \alpha_{k_1} \alpha_{k_2} \left(W_c^{k_1/2N} \right)^2 \left(W_s^{k_2/2N} \right)^2 \left[C(\mathbf{k}, \mathbf{n}) + \frac{W_c^{k_2/2N}}{W_s^{k_2/2N}} C(k_1, k'_2, \mathbf{n}) \right. \\
&\quad \left. - \frac{W_s^{k_1/2N}}{W_c^{k_1/2N}} C(k'_1, k_2, \mathbf{n}) - \frac{W_s^{k_1/2N} W_c^{k_2/2N}}{W_c^{k_1/2N} W_s^{k_2/2N}} C(\mathbf{k}', \mathbf{n}) \right],
\end{aligned} \tag{A.9}$$

$$\begin{aligned}
& 4\alpha_{k_1}\alpha_{k_2}W_s^{k_1/2N}W_c^{k_2/2N}C(2,\mathbf{k},\mathbf{n}) \\
&= \alpha_{k_1}\alpha_{k_2}\left(W_s^{k_1/2N}\right)^2\left(W_c^{k_2/2N}\right)^2\left[C(\mathbf{k},\mathbf{n})-\frac{W_s^{k_2/2N}}{W_c^{k_2/2N}}C(k_1,N-k_2,\mathbf{n})\right. \\
&\quad \left.+\frac{W_c^{k_1/2N}}{W_s^{k_1/2N}}C(N-k_1,k_2,\mathbf{n})-\frac{W_c^{k_1/2N}W_s^{k_2/2N}}{W_s^{k_1/2N}W_c^{k_2/2N}}C(N-k_1,N-k_2,\mathbf{n})\right],
\end{aligned} \tag{A.10}$$

and

$$\begin{aligned}
& 4\alpha_{k_1}\alpha_{k_2}W_s^{k_1/2N}W_s^{k_2/2N}C(3,\mathbf{k},\mathbf{n}) \\
&= \alpha_{k_1}\alpha_{k_2}\left(W_s^{k_1/2N}\right)^2\left(W_s^{k_2/2N}\right)^2\left[C(\mathbf{k},\mathbf{n})+\frac{W_c^{k_2/2N}}{W_s^{k_2/2N}}C(k_1,k'_2,\mathbf{n})\right. \\
&\quad \left.+\frac{W_c^{k_1/2N}}{W_s^{k_1/2N}}C(k'_1,k_2,\mathbf{n})+\frac{W_c^{k_1/2N}W_c^{k_2/2N}}{W_s^{k_1/2N}W_s^{k_2/2N}}C(\mathbf{k}',\mathbf{n})\right].
\end{aligned} \tag{A.11}$$

Using the results obtained for the aliased components for the four subband transforms in SBDCT, we obtain the aliased components for the seven approximate DCT algorithms.

$$\begin{aligned}
& C^{\text{ApD1}}(\mathbf{k},\mathbf{n})/\alpha_{k_1}\alpha_{k_2} \\
&= 4W_c^{k_1/2N}W_c^{k_2/2N}C(0,\mathbf{k},\mathbf{n}) \\
&= \left(W_c^{k_1/2N}\right)^2\left(W_c^{k_2/2N}\right)^2C(\mathbf{k},\mathbf{n})-\left(W_c^{k_1/2N}\right)^2W_c^{k_2/2N}W_s^{k_2/2N}C(k_1,k'_2,\mathbf{n}) \\
&\quad -W_c^{k_1/2N}W_s^{k_1/2N}\left(W_c^{k_2/2N}\right)^2C(k'_1,k_2,\mathbf{n})+W_c^{k_1/2N}W_c^{k_2/2N}W_s^{k_1/2N}W_s^{k_2/2N}C(\mathbf{k}',\mathbf{n}),
\end{aligned} \tag{A.12}$$

$$\begin{aligned}
& C^{\text{ApD2}}(\mathbf{k},\mathbf{n})/\alpha_{k_1}\alpha_{k_2} \\
&= 4W_c^{k_1/2N}W_c^{k_2/2N}C(0,\mathbf{k},\mathbf{n})+4W_c^{k_1/2N}W_s^{k_2/2N}C(1,\mathbf{k},\mathbf{n}) \\
&= \left[\left(W_c^{k_1/2N}\right)^2\left(W_c^{k_2/2N}\right)^2+\left(W_c^{k_1/2N}\right)^2\left(W_s^{k_2/2N}\right)^2\right]C(\mathbf{k},\mathbf{n}) \\
&\quad -\left[W_c^{k_1/2N}W_s^{k_1/2N}\left(W_c^{k_2/2N}\right)^2+W_c^{k_1/2N}W_s^{k_1/2N}\left(W_s^{k_2/2N}\right)^2\right]C(k'_1,k_2,\mathbf{n}) \\
&= \left(W_c^{k_1/2N}\right)^2C(\mathbf{k},\mathbf{n})-W_c^{k_1/2N}W_s^{k_1/2N}C(k'_1,k_2,\mathbf{n}),
\end{aligned} \tag{A.13}$$

$$\begin{aligned}
& C^{\text{ApD3}}(\mathbf{k},\mathbf{n})/\alpha_{k_1}\alpha_{k_2} \\
&= 4W_c^{k_1/2N}W_c^{k_2/2N}C(0,\mathbf{k},\mathbf{n})+4W_s^{k_1/2N}W_c^{k_2/2N}C(2,\mathbf{k},\mathbf{n}) \\
&= \left[\left(W_c^{k_1/2N}\right)^2\left(W_c^{k_2/2N}\right)^2+\left(W_s^{k_1/2N}\right)^2\left(W_c^{k_2/2N}\right)^2\right]C(\mathbf{k},\mathbf{n}) \\
&\quad -\left[\left(W_c^{k_1/2N}\right)^2W_c^{k_2/2N}W_s^{k_2/2N}+\left(W_s^{k_1/2N}\right)^2W_c^{k_2/2N}W_s^{k_2/2N}\right]C(k_1,k'_2,\mathbf{n}) \\
&= \left(W_c^{k_2/2N}\right)^2C(\mathbf{k},\mathbf{n})-W_c^{k_2/2N}W_s^{k_2/2N}C(k_1,k'_2,\mathbf{n}),
\end{aligned} \tag{A.14}$$

$$\begin{aligned}
& C^{\text{ApD4}}(\mathbf{k}, \mathbf{n}) / \alpha_{k_1} \alpha_{k_2} \\
&= 4W_c^{k_1/2N} W_c^{k_2/2N} C(0, \mathbf{k}, \mathbf{n}) + 4W_s^{k_1/2N} W_s^{k_2/2N} C(3, \mathbf{k}, \mathbf{n}) \\
&= \left[\left(W_c^{k_1/2N} \right)^2 \left(W_c^{k_2/2N} \right)^2 + \left(W_s^{k_1/2N} \right)^2 \left(W_s^{k_2/2N} \right)^2 \right] C(\mathbf{k}, \mathbf{n}) \\
&\quad - \left[\left(W_c^{k_1/2N} \right)^2 W_c^{k_2/2N} W_s^{k_2/2N} - \left(W_s^{k_1/2N} \right)^2 W_c^{k_2/2N} W_s^{k_2/2N} \right] C(k_1, k'_2, \mathbf{n}) \\
&\quad - \left[W_c^{k_1/2N} W_s^{k_1/2N} \left(W_c^{k_2/2N} \right)^2 - W_c^{k_1/2N} W_s^{k_1/2N} \left(W_s^{k_2/2N} \right)^2 \right] C(k'_1, k_2, \mathbf{n}) \\
&\quad + \left(W_c^{k_1/2N} W_c^{k_2/2N} W_s^{k_1/2N} W_s^{k_2/2N} + W_c^{k_1/2N} W_c^{k_2/2N} W_s^{k_1/2N} W_s^{k_2/2N} \right) C(\mathbf{k}', \mathbf{n}) \\
&= \left[\left(W_c^{k_1/2N} \right)^2 \left(W_c^{k_2/2N} \right)^2 + \left(W_s^{k_1/2N} \right)^2 \left(W_s^{k_2/2N} \right)^2 \right] C(\mathbf{k}, \mathbf{n}) \\
&\quad - \left[\left(W_c^{k_1/2N} \right)^2 - \left(W_s^{k_1/2N} \right)^2 \right] W_c^{k_2/2N} W_s^{k_2/2N} C(k_1, k'_2, \mathbf{n}) \\
&\quad - C_{2N}^{k_1} S_{2N}^{k_1} \left[\left(W_c^{k_2/2N} \right)^2 - \left(W_s^{k_2/2N} \right)^2 \right] C(k'_1, k_2, \mathbf{n}) \\
&\quad + \left(2W_c^{k_1/2N} W_c^{k_2/2N} W_s^{k_1/2N} W_s^{k_2/2N} \right) C(\mathbf{k}', \mathbf{n}) \tag{A.15}
\end{aligned}$$

$$\begin{aligned}
& C^{\text{ApD5}}(\mathbf{k}, \mathbf{n}) / \alpha_{k_1} \alpha_{k_2} \\
&= 4W_c^{k_1/2N} W_c^{k_2/2N} C(0, \mathbf{k}, \mathbf{n}) + 4W_c^{k_1/2N} W_s^{k_2/2N} C(1, \mathbf{k}, \mathbf{n}) + 4W_s^{k_1/2N} W_s^{k_2/2N} C(2, \mathbf{k}, \mathbf{n}) \\
&= \left[\left(W_c^{k_1/2N} \right)^2 \left(W_c^{k_2/2N} \right)^2 + \left(W_c^{k_1/2N} \right)^2 \left(W_s^{k_2/2N} \right)^2 + \left(W_s^{k_1/2N} \right)^2 \left(W_c^{k_2/2N} \right)^2 \right] C(\mathbf{k}, \mathbf{n}) \\
&\quad - \left[\left(W_c^{k_1/2N} \right)^2 W_c^{k_2/2N} W_s^{k_2/2N} - \left(W_c^{k_1/2N} \right)^2 W_c^{k_2/2N} W_s^{k_2/2N} + \left(W_s^{k_1/2N} \right)^2 W_c^{k_2/2N} W_s^{k_2/2N} \right] C(k_1, k'_2, \mathbf{n}) \\
&\quad - \left[W_c^{k_1/2N} W_s^{k_1/2N} \left(W_c^{k_2/2N} \right)^2 + W_c^{k_1/2N} W_s^{k_1/2N} \left(W_c^{k_2/2N} \right)^2 - W_c^{k_1/2N} W_s^{k_1/2N} \left(W_c^{k_2/2N} \right)^2 \right] C(k'_1, k_2, \mathbf{n}) \\
&\quad + \left(W_c^{k_1/2N} W_c^{k_2/2N} W_s^{k_1/2N} W_s^{k_2/2N} - W_c^{k_1/2N} W_c^{k_2/2N} W_s^{k_1/2N} W_s^{k_2/2N} - W_c^{k_1/2N} W_c^{k_2/2N} W_s^{k_1/2N} W_s^{k_2/2N} \right) C(\mathbf{k}, \mathbf{n}) \\
&= \left\{ \left(W_c^{k_1/2N} \right)^2 + \left(W_s^{k_1/2N} \right)^2 \left(W_c^{k_2/2N} \right)^2 \right\} C(\mathbf{k}, \mathbf{n}) - \left(W_s^{k_1/2N} \right)^2 W_c^{k_2/2N} W_s^{k_2/2N} C(k_1, k'_2, \mathbf{n}) \\
&\quad - W_c^{k_1/2N} W_s^{k_1/2N} \left(W_s^{k_2/2N} \right)^2 C(k'_1, k_2, \mathbf{n}) - W_c^{k_1/2N} W_c^{k_2/2N} W_s^{k_1/2N} W_s^{k_2/2N} C(\mathbf{k}', \mathbf{n}), \tag{A.16}
\end{aligned}$$

$$\begin{aligned}
& C^{\text{ApD6}}(\mathbf{k}, \mathbf{n}) / \alpha_{k_1} \alpha_{k_2} \\
&= 4W_c^{k_1/2N} W_c^{k_2/2N} C(0, \mathbf{k}, \mathbf{n}) + 4W_c^{k_1/2N} W_s^{k_2/2N} C(1, \mathbf{k}, \mathbf{n}) + 4W_s^{k_1/2N} W_s^{k_2/2N} C(3, \mathbf{k}, \mathbf{n}) \\
&= \left((W_c^{k_1/2N})^2 (W_c^{k_2/2N})^2 + (W_c^{k_1/2N})^2 (W_s^{k_2/2N})^2 + (W_s^{k_1/2N})^2 (W_s^{k_2/2N})^2 \right) C(\mathbf{k}, \mathbf{n}) \\
&\quad - \left((W_c^{k_1/2N})^2 W_c^{k_2/2N} W_s^{k_2/2N} - (W_c^{k_1/2N})^2 W_c^{k_2/2N} W_s^{k_2/2N} - (W_s^{k_1/2N})^2 W_c^{k_2/2N} W_s^{k_2/2N} \right) C(k_1, k'_2, \mathbf{n}) \\
&\quad - \left(W_c^{k_1/2N} W_s^{k_1/2N} (W_c^{k_2/2N})^2 + W_c^{k_1/2N} W_s^{k_1/2N} (W_s^{k_2/2N})^2 - W_c^{k_1/2N} W_s^{k_1/2N} (W_s^{k_2/2N})^2 \right) C(k'_1, k_2, \mathbf{n}) \\
&\quad + \left(W_c^{k_1/2N} W_c^{k_2/2N} W_s^{k_1/2N} W_s^{k_2/2N} - W_c^{k_1/2N} W_c^{k_2/2N} W_s^{k_1/2N} W_s^{k_2/2N} + W_c^{k_1/2N} W_c^{k_2/2N} W_s^{k_1/2N} W_s^{k_2/2N} \right) C(\mathbf{k}', \mathbf{n}) \\
&= \left\{ (W_c^{k_1/2N})^2 + (W_s^{k_1/2N})^2 (W_s^{k_2/2N})^2 \right\} C(\mathbf{k}, \mathbf{n}) + (W_s^{k_1/2N})^2 W_c^{k_2/2N} W_s^{k_2/2N} C(k_1, k'_2, \mathbf{n}) \\
&\quad - W_c^{k_1/2N} W_s^{k_1/2N} (W_c^{k_2/2N})^2 C(k'_1, k_2, \mathbf{n}) + W_c^{k_1/2N} W_c^{k_2/2N} W_s^{k_1/2N} W_s^{k_2/2N} C(\mathbf{k}', \mathbf{n}), \tag{A.17}
\end{aligned}$$

and

$$\begin{aligned}
& C^{\text{ApD7}}(\mathbf{k}, \mathbf{n}) / \alpha_{k_1} \alpha_{k_2} \\
&= 4W_c^{k_1/2N} W_c^{k_2/2N} C(0, \mathbf{k}, \mathbf{n}) + 4W_s^{k_1/2N} W_c^{k_2/2N} C(2, \mathbf{k}, \mathbf{n}) + 4W_s^{k_1/2N} W_s^{k_2/2N} C(3, \mathbf{k}, \mathbf{n}) \\
&= \left((W_c^{k_1/2N})^2 (W_c^{k_2/2N})^2 + (W_s^{k_1/2N})^2 (W_c^{k_2/2N})^2 + (W_s^{k_1/2N})^2 (W_s^{k_2/2N})^2 \right) C(\mathbf{k}, \mathbf{n}) \\
&\quad - \left((W_c^{k_1/2N})^2 W_c^{k_2/2N} W_s^{k_2/2N} + (W_s^{k_1/2N})^2 W_c^{k_2/2N} W_s^{k_2/2N} - (W_s^{k_1/2N})^2 W_c^{k_2/2N} W_s^{k_2/2N} \right) C(k_1, k'_2, \mathbf{n}) \\
&\quad - \left(W_c^{k_1/2N} W_s^{k_1/2N} (W_c^{k_2/2N})^2 - W_c^{k_1/2N} W_s^{k_1/2N} (W_c^{k_2/2N})^2 - W_c^{k_1/2N} W_s^{k_1/2N} (W_s^{k_2/2N})^2 \right) C(k'_1, k_2, \mathbf{n}) \\
&\quad + \left(W_c^{k_1/2N} W_c^{k_2/2N} W_s^{k_1/2N} W_s^{k_2/2N} - W_c^{k_1/2N} W_c^{k_2/2N} W_s^{k_1/2N} W_s^{k_2/2N} + W_c^{k_1/2N} W_c^{k_2/2N} W_s^{k_1/2N} W_s^{k_2/2N} \right) C(\mathbf{k}', \mathbf{n}) \\
&= \left\{ (W_c^{k_2/2N})^2 + (W_s^{k_1/2N})^2 (W_s^{k_2/2N})^2 \right\} C(\mathbf{k}, \mathbf{n}) - (W_c^{k_1/2N})^2 W_c^{k_2/2N} W_s^{k_2/2N} C(k_1, k'_2, \mathbf{n}) \\
&\quad + W_c^{k_1/2N} W_s^{k_1/2N} (W_s^{k_2/2N})^2 C(k'_1, k_2, \mathbf{n}) + W_c^{k_1/2N} W_c^{k_2/2N} W_s^{k_1/2N} W_s^{k_2/2N} C(\mathbf{k}', \mathbf{n}). \tag{A.18}
\end{aligned}$$

Appendix B

Test-Images

The following are 20 test-images, which are frequently found in literature and are downloaded from <http://decsai.ugr.es/cvg/dbimagenes/g512.php>. These test-images are of 512×512 .

TI-1

TI-2

TI-3

TI-4

TI-5

TI-6

TI-7

TI-8

TI-9

TI-10

TI-11

TI-12

TI-13

TI-14

TI-15

TI-16

TI-17

TI-18

TI-19

TI-20

The following is 12 test-images, which are obtained from <http://www.cipr.rpi.edu/resource/stills/canon.html>. Test-images TI-21 to TI-23 are of 480×512 , and test-images TI-24 to TI-32 are of 512×480 .

TI-21

TI-22

TI-23

TI-24

TI-25

TI-26

TI-27

TI-28

TI-29

TI-30

TI-31

TI-32

The following is 13 test-images, which are obtained from <http://www.cipr.rpi.edu/resource/stills/kodak.html>. Test-images TI-23 to TI-45 are of 768×512 .

TI-33

TI-34

TI-35

TI-36

TI-37

TI-38

TI-39

TI-40

TI-41

TI-42

TI-43

TI-44

TI-45

Appendix C

PSNR and MSSIM values

Table C.1 PSNR values for downsizing by factor of 2/3 using various methods.

Test Image	PSNR (dB)						
	prop($\tilde{N}$)			Park [PP06]		Shu [SC07]	
	$\tilde{N} = N$	$\tilde{N} = N_{0.05}$	$\tilde{N} = N_{PT}$	park(I)	park(II)	shu(I)	shu(II)
TI-1	24.45	24.44	24.43	24.43	24.27	24.45	24.45
TI-2	28.87	28.86	28.84	28.86	28.78	28.87	28.87
TI-3	32.38	32.38	32.26	32.37	32.19	32.38	32.38
TI-4	33.70	33.70	33.56	33.71	33.51	33.70	33.70
TI-5	35.93	35.92	35.83	36.01	35.88	35.93	35.93
TI-6	38.74	38.74	38.36	38.76	38.59	38.74	38.74
TI-7	38.19	38.18	38.04	38.27	38.17	38.19	38.19
TI-8	38.47	38.47	38.10	38.48	38.32	38.47	38.47
TI-9	36.20	36.20	36.10	36.29	36.13	36.20	36.20
TI-10	30.29	30.28	30.21	30.26	30.05	30.29	30.29
TI-11	31.01	31.01	30.92	31.01	30.84	31.01	31.01
TI-12	33.66	33.65	33.50	33.68	33.54	33.66	33.66
TI-13	25.73	25.72	25.69	25.73	25.54	25.73	25.73
TI-14	35.31	35.31	35.09	35.32	35.14	35.31	35.31
TI-15	28.56	28.55	28.50	28.55	28.37	28.56	28.56
TI-16	32.61	32.60	32.51	32.60	32.42	32.61	32.61
TI-17	30.04	30.04	29.97	30.04	29.87	30.04	30.04
TI-18	27.15	27.14	27.11	27.15	26.97	27.15	27.15
TI-19	34.08	34.07	33.90	34.09	33.96	34.08	34.08
TI-20	39.11	39.11	38.97	39.35	39.24	39.11	39.11
TI-21	48.09	48.08	46.75	48.15	47.97	48.09	48.09
TI-22	46.93	46.93	45.85	47.05	46.85	46.93	46.93
TI-23	44.71	44.71	43.88	44.80	44.61	44.71	44.71
TI-24	40.72	40.72	40.41	40.76	40.58	40.72	40.72
TI-25	38.77	38.76	38.53	38.82	38.61	38.77	38.77
TI-26	46.86	46.85	45.86	46.97	46.73	46.86	46.86
TI-27	45.19	45.19	44.39	45.19	44.99	45.19	45.19
TI-28	41.34	41.33	40.99	41.37	41.16	41.34	41.34
TI-29	46.13	46.13	45.31	46.21	46.03	46.13	46.13
TI-30	45.17	45.17	44.39	45.32	45.10	45.17	45.17
TI-31	42.33	42.32	41.83	42.48	42.24	42.33	42.33
TI-32	40.93	40.92	40.64	41.02	40.81	40.93	40.93
TI-33	29.81	29.81	29.74	29.78	29.65	29.81	29.81
TI-34	36.80	36.79	36.55	36.76	36.63	36.80	36.80
TI-35	38.50	38.49	38.24	38.48	38.35	38.50	38.50
TI-36	27.36	27.35	27.31	27.34	27.20	27.36	27.36
TI-37	31.17	31.16	31.08	31.15	31.02	31.17	31.17
TI-38	31.09	31.09	31.01	31.08	30.95	31.09	31.09
TI-39	38.36	38.35	38.10	38.36	38.22	38.36	38.36
TI-40	27.18	27.18	27.14	27.16	27.01	27.18	27.18
TI-41	34.38	34.37	34.22	34.35	34.22	34.38	34.38
TI-42	32.35	32.34	32.24	32.32	32.18	32.35	32.35
TI-43	32.80	32.80	32.70	32.78	32.65	32.80	32.80
TI-44	36.72	36.72	36.51	36.72	36.59	36.72	36.72
TI-45	33.42	33.41	33.27	33.39	33.25	33.42	33.42

Table C.2 PSNR values for downsizing by factor of 3/4 using various methods.

Test Image	PSNR (dB)						
	prop($\tilde{N}$)			Park [PP06]		Shu [SC07]	
	$\tilde{N} = N$	$\tilde{N} = N_{0.05}$	$\tilde{N} = N_{PT}$	park(I)	park(II)	shu(I)	shu(II)
TI-1	26.21	26.18	26.16	26.24	25.42	26.21	26.21
TI-2	31.41	31.39	31.31	31.54	31.02	31.41	31.41
TI-3	33.94	33.91	33.71	33.96	33.16	33.94	33.94
TI-4	35.15	35.12	34.90	35.20	34.38	35.15	35.15
TI-5	37.05	37.03	36.86	37.30	36.67	37.05	37.05
TI-6	40.62	40.59	39.97	40.71	39.78	40.62	40.62
TI-7	40.44	40.42	40.12	40.67	40.00	40.44	40.44
TI-8	39.87	39.84	39.27	39.90	39.22	39.87	39.87
TI-9	38.00	37.97	37.76	38.23	37.42	38.00	38.00
TI-10	32.00	31.97	31.83	32.05	31.02	32.00	32.00
TI-11	32.78	32.75	32.58	32.82	32.01	32.78	32.78
TI-12	35.20	35.17	34.90	35.26	34.54	35.20	35.20
TI-13	27.01	26.98	26.93	27.02	26.22	27.01	27.01
TI-14	37.18	37.15	36.76	37.26	36.38	37.18	37.18
TI-15	29.96	29.93	29.83	29.97	29.17	29.96	29.96
TI-16	34.49	34.46	34.28	34.54	33.76	34.49	34.49
TI-17	31.73	31.70	31.58	31.76	30.88	31.73	31.73
TI-18	28.72	28.69	28.64	28.79	27.99	28.72	28.72
TI-19	35.06	35.03	34.75	35.07	34.45	35.06	35.06
TI-20	40.22	40.20	39.94	40.71	40.08	40.22	40.22
TI-21	49.67	49.63	47.96	49.89	48.83	49.67	49.67
TI-22	48.81	48.76	47.29	49.10	47.82	48.81	48.81
TI-23	47.04	46.98	45.71	47.37	45.90	47.04	47.04
TI-24	43.88	43.82	43.15	44.19	42.52	43.88	43.88
TI-25	41.90	41.84	41.34	42.27	40.51	41.90	41.90
TI-26	48.79	48.73	47.32	49.12	47.73	48.79	48.79
TI-27	47.93	47.88	46.47	48.16	46.64	47.93	47.93
TI-28	44.54	44.48	43.73	44.86	43.21	44.54	44.54
TI-29	48.28	48.23	47.06	48.60	47.21	48.28	48.28
TI-30	47.19	47.13	45.85	47.62	46.17	47.19	47.19
TI-31	45.07	44.99	44.03	45.59	43.55	45.07	45.07
TI-32	44.10	44.04	43.47	44.59	42.60	44.10	44.10
TI-33	32.26	32.23	32.09	32.28	31.51	32.26	32.26
TI-34	39.06	39.03	38.61	39.09	38.32	39.06	39.06
TI-35	40.72	40.69	40.25	40.79	40.04	40.72	40.72
TI-36	29.47	29.43	29.34	29.51	28.78	29.47	29.47
TI-37	34.02	33.98	33.77	34.11	33.22	34.02	34.02
TI-38	33.26	33.22	33.08	33.31	32.58	33.26	33.26
TI-39	40.75	40.71	40.24	40.86	39.89	40.75	40.75
TI-40	29.25	29.22	29.16	29.29	28.54	29.25	29.25
TI-41	36.70	36.66	36.36	36.73	35.95	36.70	36.70
TI-42	34.46	34.43	34.25	34.50	33.71	34.46	34.46
TI-43	35.07	35.04	34.84	35.11	34.34	35.07	35.07
TI-44	38.49	38.44	38.13	38.53	37.91	38.49	38.49
TI-45	35.70	35.66	35.38	35.75	34.93	35.70	35.70

Table C.3 PSNR values for downsizing by factor of 4/5 using various methods.

Test Image	PSNR (dB)						
	prop($\tilde{N}$)			Park [PP06]		Shu [SC07]	
	$\tilde{N} = N$	$\tilde{N} = N_{0.05}$	$\tilde{N} = N_{PT}$	park(I)	park(II)	shu(I)	shu(II)
TI-1	27.22	27.22	27.12	27.31	27.31	27.22	27.22
TI-2	33.16	33.16	32.88	33.30	33.30	33.16	33.16
TI-3	34.87	34.86	34.49	34.89	34.89	34.87	34.87
TI-4	35.94	35.94	35.60	36.04	36.04	35.94	35.94
TI-5	38.75	38.75	38.23	39.47	39.47	38.75	38.75
TI-6	41.62	41.62	40.76	41.80	41.80	41.62	41.62
TI-7	41.54	41.54	41.06	42.03	42.03	41.54	41.54
TI-8	40.73	40.73	40.07	40.74	40.74	40.73	40.73
TI-9	38.41	38.40	38.00	38.83	38.83	38.41	38.41
TI-10	32.46	32.46	32.12	32.57	32.57	32.46	32.46
TI-11	34.14	34.14	33.72	34.21	34.21	34.14	34.14
TI-12	36.34	36.33	35.87	36.40	36.40	36.34	36.34
TI-13	28.24	28.23	28.05	28.25	28.25	28.24	28.24
TI-14	38.15	38.14	37.54	38.27	38.27	38.15	38.15
TI-15	30.99	30.99	30.74	31.01	31.01	30.99	30.99
TI-16	35.55	35.55	35.17	35.61	35.61	35.55	35.55
TI-17	32.57	32.57	32.29	32.63	32.63	32.57	32.57
TI-18	29.71	29.71	29.51	29.86	29.86	29.71	29.71
TI-19	35.86	35.86	35.54	35.85	35.85	35.86	35.86
TI-20	41.39	41.39	40.84	42.52	42.52	41.39	41.39
TI-21	50.51	50.50	48.57	50.90	50.90	50.51	50.51
TI-22	49.70	49.70	47.90	50.26	50.26	49.70	49.70
TI-23	48.13	48.12	46.40	48.76	48.76	48.13	48.13
TI-24	45.20	45.19	43.85	45.92	45.92	45.20	45.20
TI-25	43.14	43.13	41.88	44.03	44.03	43.14	43.14
TI-26	49.68	49.67	47.94	50.31	50.31	49.68	49.68
TI-27	48.94	48.93	47.07	49.48	49.48	48.94	48.94
TI-28	45.68	45.67	44.33	46.47	46.47	45.68	45.68
TI-29	49.23	49.23	47.71	49.83	49.83	49.23	49.23
TI-30	48.26	48.25	46.68	49.01	49.01	48.26	48.26
TI-31	46.11	46.10	44.45	47.18	47.18	46.11	46.11
TI-32	45.48	45.47	44.20	46.47	46.47	45.48	45.48
TI-33	33.74	33.74	33.31	33.81	33.81	33.74	33.74
TI-34	40.39	40.38	39.62	40.43	40.43	40.39	40.39
TI-35	42.28	42.27	41.39	42.40	42.40	42.28	42.28
TI-36	30.74	30.74	30.41	30.78	30.78	30.74	30.74
TI-37	35.68	35.67	35.03	35.88	35.88	35.68	35.68
TI-38	34.27	34.26	33.91	34.35	34.35	34.27	34.27
TI-39	42.39	42.39	41.44	42.57	42.57	42.39	42.39
TI-40	30.53	30.53	30.33	30.60	30.60	30.53	30.53
TI-41	37.86	37.86	37.26	37.91	37.91	37.86	37.86
TI-42	35.92	35.92	35.45	35.97	35.97	35.92	35.92
TI-43	36.31	36.31	35.83	36.38	36.38	36.31	36.31
TI-44	39.65	39.65	39.06	39.75	39.75	39.65	39.65
TI-45	37.18	37.17	36.52	37.27	37.27	37.18	37.18

Table C.4 PSNR values for downsizing by factor of 5/6 using various methods.

Test Image	PSNR (dB)						
	prop($\tilde{N}$)			Park [PP06]		Shu [SC07]	
	$\tilde{N} = N$	$\tilde{N} = N_{0.05}$	$\tilde{N} = N_{PT}$	park(I)	park(II)	shu(I)	shu(II)
TI-1	27.22	27.22	27.12	27.31	27.31	27.22	27.22
TI-2	33.16	33.16	32.88	33.30	33.30	33.16	33.16
TI-3	34.87	34.86	34.49	34.89	34.89	34.87	34.87
TI-4	35.94	35.94	35.60	36.04	36.04	35.94	35.94
TI-5	38.75	38.75	38.23	39.47	39.47	38.75	38.75
TI-6	41.62	41.62	40.76	41.80	41.80	41.62	41.62
TI-7	41.54	41.54	41.06	42.03	42.03	41.54	41.54
TI-8	40.73	40.73	40.07	40.74	40.74	40.73	40.73
TI-9	38.41	38.40	38.00	38.83	38.83	38.41	38.41
TI-10	32.46	32.46	32.12	32.57	32.57	32.46	32.46
TI-11	34.14	34.14	33.72	34.21	34.21	34.14	34.14
TI-12	36.34	36.33	35.87	36.40	36.40	36.34	36.34
TI-13	28.24	28.23	28.05	28.25	28.25	28.24	28.24
TI-14	38.15	38.14	37.54	38.27	38.27	38.15	38.15
TI-15	30.99	30.99	30.74	31.01	31.01	30.99	30.99
TI-16	35.55	35.55	35.17	35.61	35.61	35.55	35.55
TI-17	32.57	32.57	32.29	32.63	32.63	32.57	32.57
TI-18	29.71	29.71	29.51	29.86	29.86	29.71	29.71
TI-19	35.86	35.86	35.54	35.85	35.85	35.86	35.86
TI-20	41.39	41.39	40.84	42.52	42.52	41.39	41.39
TI-21	50.51	50.50	48.57	50.90	50.90	50.51	50.51
TI-22	49.70	49.70	47.90	50.26	50.26	49.70	49.70
TI-23	48.13	48.12	46.40	48.76	48.76	48.13	48.13
TI-24	45.20	45.19	43.85	45.92	45.92	45.20	45.20
TI-25	43.14	43.13	41.88	44.03	44.03	43.14	43.14
TI-26	49.68	49.67	47.94	50.31	50.31	49.68	49.68
TI-27	48.94	48.93	47.07	49.48	49.48	48.94	48.94
TI-28	45.68	45.67	44.33	46.47	46.47	45.68	45.68
TI-29	49.23	49.23	47.71	49.83	49.83	49.23	49.23
TI-30	48.26	48.25	46.68	49.01	49.01	48.26	48.26
TI-31	46.11	46.10	44.45	47.18	47.18	46.11	46.11
TI-32	45.48	45.47	44.20	46.47	46.47	45.48	45.48
TI-33	33.74	33.74	33.31	33.81	33.81	33.74	33.74
TI-34	40.39	40.38	39.62	40.43	40.43	40.39	40.39
TI-35	42.28	42.27	41.39	42.40	42.40	42.28	42.28
TI-36	30.74	30.74	30.41	30.78	30.78	30.74	30.74
TI-37	35.68	35.67	35.03	35.88	35.88	35.68	35.68
TI-38	34.27	34.26	33.91	34.35	34.35	34.27	34.27
TI-39	42.39	42.39	41.44	42.57	42.57	42.39	42.39
TI-40	30.53	30.53	30.33	30.60	30.60	30.53	30.53
TI-41	37.86	37.86	37.26	37.91	37.91	37.86	37.86
TI-42	35.92	35.92	35.45	35.97	35.97	35.92	35.92
TI-43	36.31	36.31	35.83	36.38	36.38	36.31	36.31
TI-44	39.65	39.65	39.06	39.75	39.75	39.65	39.65
TI-45	37.18	37.17	36.52	37.27	37.27	37.18	37.18

Table C.5 PSNR values for downsizing by factor of 6/7 using various methods.

Test Image	PSNR (dB)						
	prop($\tilde{N}$)			Park [PP06]		Shu [SC07]	
	$\tilde{N} = N$	$\tilde{N} = N_{0.05}$	$\tilde{N} = N_{PT}$	park(I)	park(II)	shu(I)	shu(II)
TI-1	28.99	28.94	28.84	29.06	29.06	28.99	28.99
TI-2	36.28	36.19	35.77	36.62	36.62	36.28	36.28
TI-3	36.47	36.42	35.97	36.53	36.53	36.47	36.47
TI-4	37.67	37.62	37.16	37.75	37.75	37.67	37.67
TI-5	40.27	40.23	39.96	40.68	40.68	40.27	40.27
TI-6	43.32	43.26	42.05	43.51	43.51	43.32	43.32
TI-7	43.15	43.11	42.56	43.50	43.50	43.15	43.15
TI-8	42.40	42.35	41.28	42.45	42.45	42.40	42.40
TI-9	39.98	39.94	39.63	40.25	40.25	39.98	39.98
TI-10	34.38	34.33	33.99	34.48	34.48	34.38	34.38
TI-11	35.67	35.62	35.11	35.77	35.77	35.67	35.67
TI-12	38.17	38.12	37.29	38.29	38.29	38.17	38.17
TI-13	29.52	29.48	29.26	29.55	29.55	29.52	29.52
TI-14	39.96	39.90	39.28	40.12	40.12	39.96	39.96
TI-15	32.51	32.47	32.14	32.55	32.55	32.51	32.51
TI-16	37.45	37.38	36.92	37.55	37.55	37.45	37.45
TI-17	34.33	34.28	33.98	34.41	34.41	34.33	34.33
TI-18	31.39	31.35	31.19	31.50	31.50	31.39	31.39
TI-19	37.55	37.51	36.61	37.58	37.58	37.55	37.55
TI-20	42.69	42.65	42.27	43.26	43.26	42.69	42.69
TI-21	51.55	51.49	48.60	51.90	51.90	51.55	51.55
TI-22	50.72	50.66	48.14	51.24	51.24	50.72	50.72
TI-23	49.64	49.55	47.11	50.24	50.24	49.64	49.64
TI-24	47.19	47.06	45.89	47.98	47.98	47.19	47.19
TI-25	45.43	45.30	44.38	46.29	46.29	45.43	45.43
TI-26	50.81	50.74	48.43	51.42	51.42	50.81	50.81
TI-27	50.44	50.34	48.20	50.95	50.95	50.44	50.44
TI-28	47.52	47.40	46.11	48.30	48.30	47.52	47.52
TI-29	50.65	50.58	48.31	51.13	51.13	50.65	50.65
TI-30	49.37	49.29	47.51	50.04	50.04	49.37	49.37
TI-31	47.58	47.46	46.22	48.65	48.65	47.58	47.58
TI-32	47.13	47.02	45.96	48.07	48.07	47.13	47.13
TI-33	36.10	36.02	35.53	36.24	36.24	36.10	36.10
TI-34	42.59	42.52	41.47	42.71	42.71	42.59	42.59
TI-35	44.43	44.34	43.17	44.64	44.64	44.43	44.43
TI-36	32.92	32.85	32.42	33.03	33.03	32.92	32.92
TI-37	38.05	37.95	37.19	38.38	38.38	38.05	38.05
TI-38	36.54	36.47	36.08	36.67	36.67	36.54	36.54
TI-39	44.73	44.64	43.37	45.06	45.06	44.73	44.73
TI-40	32.70	32.63	32.41	32.80	32.80	32.70	32.70
TI-41	40.06	39.98	39.18	40.21	40.21	40.06	40.06
TI-42	37.96	37.89	37.28	38.08	38.08	37.96	37.96
TI-43	38.56	38.48	37.91	38.69	38.69	38.56	38.56
TI-44	41.71	41.64	40.90	41.84	41.84	41.71	41.71
TI-45	39.51	39.42	38.63	39.71	39.71	39.51	39.51

Table C.6 PSNR values for downsizing by factor of 7/8 using various methods.

Test Image	PSNR (dB)						
	prop($\tilde{N}$)			Park [PP06]		Shu [SC07]	
	$\tilde{N} = N$	$\tilde{N} = N_{0.05}$	$\tilde{N} = N_{PT}$	park(I)	park(II)	shu(I)	shu(II)
TI-1	29.89	29.86	29.72	29.95	27.91	29.89	29.89
TI-2	38.96	38.85	38.20	39.33	35.88	38.96	38.96
TI-3	37.03	37.01	36.45	37.08	35.25	37.03	37.03
TI-4	38.34	38.31	37.67	38.39	36.42	38.34	38.34
TI-5	41.46	41.42	40.68	41.77	39.81	41.46	41.46
TI-6	43.99	43.94	42.39	44.11	41.75	43.99	43.99
TI-7	44.86	44.81	43.68	45.18	43.07	44.86	44.86
TI-8	42.98	42.96	41.25	43.00	41.19	42.98	42.98
TI-9	42.24	42.20	41.17	42.53	40.11	42.24	42.24
TI-10	34.86	34.83	34.50	34.95	32.86	34.86	34.86
TI-11	36.21	36.18	35.72	36.28	34.23	36.21	36.21
TI-12	38.95	38.91	37.98	39.02	36.99	38.95	38.95
TI-13	30.00	29.97	29.79	30.02	28.25	30.00	30.00
TI-14	40.56	40.52	39.41	40.65	38.43	40.56	40.56
TI-15	33.23	33.20	32.91	33.26	31.37	33.23	33.23
TI-16	37.97	37.93	37.33	38.01	36.11	37.97	37.97
TI-17	35.10	35.07	34.72	35.15	33.18	35.10	35.10
TI-18	32.34	32.31	32.11	32.43	30.41	32.34	32.34
TI-19	38.43	38.40	37.50	38.46	36.56	38.43	38.43
TI-20	44.04	44.00	43.10	44.50	42.44	44.04	44.04
TI-21	51.96	51.92	49.13	52.22	49.88	51.96	51.96
TI-22	51.26	51.20	48.48	51.63	48.85	51.26	51.26
TI-23	50.26	50.19	47.63	50.68	47.26	50.26	50.26
TI-24	47.97	47.87	46.11	48.55	44.24	47.97	47.97
TI-25	46.38	46.27	44.97	46.99	42.43	46.38	46.38
TI-26	51.35	51.30	48.75	51.75	48.59	51.35	51.35
TI-27	50.90	50.83	47.93	51.26	47.97	50.90	50.90
TI-28	48.49	48.39	46.37	49.03	45.03	48.49	48.49
TI-29	51.12	51.07	48.99	51.49	48.69	51.12	51.12
TI-30	50.01	49.94	47.12	50.49	47.01	50.01	50.01
TI-31	48.49	48.38	46.06	49.22	44.39	48.49	48.49
TI-32	48.05	47.95	46.35	48.73	44.32	48.05	48.05
TI-33	37.38	37.32	36.72	37.48	34.88	37.38	37.38
TI-34	43.71	43.66	42.22	43.81	41.47	43.71	43.71
TI-35	45.46	45.40	43.96	45.61	43.04	45.46	45.46
TI-36	33.88	33.83	33.44	33.96	31.68	33.88	33.88
TI-37	39.25	39.17	38.32	39.49	36.36	39.25	39.25
TI-38	37.66	37.62	37.11	37.74	35.61	37.66	37.66
TI-39	45.51	45.44	43.93	45.77	42.76	45.51	45.51
TI-40	33.49	33.45	33.21	33.57	31.44	33.49	33.49
TI-41	41.36	41.30	40.18	41.45	39.06	41.36	41.36
TI-42	39.10	39.05	38.43	39.18	36.64	39.10	39.10
TI-43	39.76	39.71	38.91	39.87	37.35	39.76	39.76
TI-44	42.83	42.78	41.71	42.91	41.08	42.83	42.83
TI-45	40.62	40.55	39.43	40.76	38.13	40.62	40.62

Table C.7 MSSIM values for downsizing by factor of 2/3 using various methods.

Test Image	PSNR (dB)						
	prop($\tilde{N}$)			Park [PP06]		Shu [SC07]	
	$\tilde{N} = N$	$\tilde{N} = N_{0.05}$	$\tilde{N} = N_{PT}$	park(I)	park(II)	shu(I)	shu(II)
TI-1	0.880	0.881	0.880	0.881	0.879	0.880	0.880
TI-2	0.930	0.929	0.928	0.930	0.929	0.930	0.930
TI-3	0.922	0.922	0.917	0.922	0.919	0.922	0.922
TI-4	0.920	0.920	0.915	0.920	0.917	0.920	0.920
TI-5	0.974	0.974	0.973	0.975	0.974	0.974	0.974
TI-6	0.963	0.963	0.956	0.963	0.961	0.963	0.963
TI-7	0.977	0.977	0.975	0.978	0.977	0.977	0.977
TI-8	0.936	0.936	0.928	0.936	0.933	0.936	0.936
TI-9	0.959	0.959	0.958	0.961	0.960	0.959	0.959
TI-10	0.940	0.940	0.937	0.940	0.938	0.940	0.940
TI-11	0.915	0.915	0.912	0.915	0.912	0.915	0.915
TI-12	0.911	0.911	0.905	0.912	0.909	0.911	0.911
TI-13	0.842	0.842	0.838	0.842	0.835	0.842	0.842
TI-14	0.941	0.940	0.935	0.941	0.938	0.941	0.941
TI-15	0.889	0.888	0.884	0.889	0.884	0.889	0.889
TI-16	0.937	0.937	0.933	0.938	0.935	0.937	0.937
TI-17	0.918	0.918	0.915	0.918	0.915	0.918	0.918
TI-18	0.905	0.905	0.904	0.906	0.903	0.905	0.905
TI-19	0.853	0.852	0.845	0.853	0.849	0.853	0.853
TI-20	0.985	0.984	0.983	0.986	0.985	0.985	0.985
TI-21	0.990	0.990	0.986	0.990	0.989	0.990	0.990
TI-22	0.990	0.990	0.987	0.991	0.990	0.990	0.990
TI-23	0.987	0.987	0.982	0.987	0.986	0.987	0.987
TI-24	0.987	0.987	0.985	0.987	0.987	0.987	0.987
TI-25	0.988	0.988	0.985	0.988	0.987	0.988	0.988
TI-26	0.990	0.990	0.987	0.990	0.990	0.990	0.990
TI-27	0.990	0.990	0.986	0.990	0.989	0.990	0.990
TI-28	0.987	0.987	0.985	0.987	0.987	0.987	0.987
TI-29	0.989	0.989	0.985	0.989	0.988	0.989	0.989
TI-30	0.991	0.991	0.988	0.991	0.990	0.991	0.991
TI-31	0.991	0.991	0.989	0.991	0.991	0.991	0.991
TI-32	0.987	0.987	0.985	0.987	0.987	0.987	0.987
TI-33	0.908	0.908	0.905	0.908	0.905	0.908	0.908
TI-34	0.944	0.944	0.937	0.944	0.942	0.944	0.944
TI-35	0.969	0.969	0.964	0.969	0.968	0.969	0.969
TI-36	0.896	0.896	0.893	0.896	0.893	0.896	0.896
TI-37	0.952	0.952	0.950	0.952	0.951	0.952	0.952
TI-38	0.923	0.923	0.919	0.923	0.921	0.923	0.923
TI-39	0.979	0.979	0.976	0.979	0.979	0.979	0.979
TI-40	0.907	0.907	0.904	0.906	0.904	0.907	0.907
TI-41	0.945	0.945	0.940	0.945	0.943	0.945	0.945
TI-42	0.948	0.948	0.943	0.948	0.947	0.948	0.948
TI-43	0.933	0.933	0.928	0.932	0.930	0.933	0.933
TI-44	0.953	0.953	0.947	0.952	0.951	0.953	0.953
TI-45	0.944	0.944	0.940	0.944	0.942	0.944	0.944

Table C.8 MSSIM values for downsizing by factor of 3/4 using various methods.

Test Image	PSNR (dB)						
	prop($\tilde{N}$)			Park [PP06]		Shu [SC07]	
	$\tilde{N} = N$	$\tilde{N} = N_{0.05}$	$\tilde{N} = N_{PT}$	park(I)	park(II)	shu(I)	shu(II)
TI-1	0.912	0.912	0.911	0.915	0.912	0.912	0.912
TI-2	0.958	0.957	0.955	0.960	0.957	0.958	0.958
TI-3	0.943	0.942	0.937	0.943	0.934	0.943	0.943
TI-4	0.938	0.937	0.931	0.938	0.927	0.938	0.938
TI-5	0.981	0.980	0.979	0.982	0.981	0.981	0.981
TI-6	0.973	0.973	0.967	0.973	0.968	0.973	0.973
TI-7	0.984	0.984	0.982	0.986	0.985	0.984	0.984
TI-8	0.951	0.951	0.942	0.951	0.943	0.951	0.951
TI-9	0.973	0.973	0.970	0.975	0.971	0.973	0.973
TI-10	0.958	0.958	0.955	0.959	0.951	0.958	0.958
TI-11	0.943	0.942	0.938	0.943	0.934	0.943	0.943
TI-12	0.932	0.931	0.924	0.932	0.923	0.932	0.932
TI-13	0.889	0.888	0.883	0.890	0.872	0.889	0.889
TI-14	0.958	0.958	0.952	0.959	0.951	0.958	0.958
TI-15	0.919	0.918	0.913	0.920	0.906	0.919	0.919
TI-16	0.957	0.957	0.952	0.958	0.951	0.957	0.957
TI-17	0.941	0.941	0.937	0.942	0.932	0.941	0.941
TI-18	0.936	0.935	0.933	0.938	0.928	0.936	0.936
TI-19	0.885	0.883	0.873	0.885	0.869	0.885	0.885
TI-20	0.988	0.987	0.986	0.989	0.989	0.988	0.988
TI-21	0.992	0.992	0.989	0.993	0.991	0.992	0.992
TI-22	0.993	0.993	0.990	0.993	0.992	0.993	0.993
TI-23	0.991	0.991	0.987	0.991	0.989	0.991	0.991
TI-24	0.993	0.993	0.990	0.993	0.991	0.993	0.993
TI-25	0.993	0.993	0.991	0.993	0.991	0.993	0.993
TI-26	0.993	0.993	0.990	0.993	0.992	0.993	0.993
TI-27	0.994	0.994	0.991	0.994	0.992	0.994	0.994
TI-28	0.993	0.993	0.990	0.993	0.991	0.993	0.993
TI-29	0.992	0.992	0.989	0.992	0.991	0.992	0.992
TI-30	0.994	0.993	0.991	0.994	0.992	0.994	0.994
TI-31	0.995	0.995	0.993	0.995	0.993	0.995	0.995
TI-32	0.993	0.993	0.991	0.993	0.991	0.993	0.993
TI-33	0.945	0.944	0.940	0.946	0.938	0.945	0.945
TI-34	0.966	0.966	0.959	0.966	0.961	0.966	0.966
TI-35	0.980	0.980	0.976	0.980	0.978	0.980	0.980
TI-36	0.938	0.937	0.933	0.939	0.929	0.938	0.938
TI-37	0.974	0.974	0.971	0.974	0.971	0.974	0.974
TI-38	0.954	0.953	0.949	0.955	0.948	0.954	0.954
TI-39	0.987	0.987	0.983	0.987	0.985	0.987	0.987
TI-40	0.943	0.943	0.939	0.944	0.937	0.943	0.943
TI-41	0.967	0.967	0.961	0.967	0.962	0.967	0.967
TI-42	0.967	0.967	0.962	0.968	0.963	0.967	0.967
TI-43	0.958	0.958	0.953	0.959	0.953	0.958	0.958
TI-44	0.970	0.970	0.964	0.970	0.966	0.970	0.970
TI-45	0.966	0.966	0.961	0.967	0.962	0.966	0.966

Table C.9 MSSIM values for downsizing by factor of 4/5 using various methods.

Test Image	PSNR (dB)						
	prop($\tilde{N}$)			Park [PP06]		Shu [SC07]	
	$\tilde{N} = N$	$\tilde{N} = N_{0.05}$	$\tilde{N} = N_{PT}$	park(I)	park(II)	shu(I)	shu(II)
TI-1	0.935	0.935	0.934	0.937	0.937	0.935	0.935
TI-2	0.972	0.972	0.969	0.974	0.974	0.972	0.972
TI-3	0.955	0.955	0.950	0.956	0.956	0.955	0.955
TI-4	0.946	0.946	0.940	0.947	0.947	0.946	0.946
TI-5	0.986	0.986	0.984	0.989	0.989	0.986	0.986
TI-6	0.979	0.979	0.973	0.979	0.979	0.979	0.979
TI-7	0.986	0.986	0.984	0.988	0.988	0.986	0.986
TI-8	0.961	0.961	0.954	0.961	0.961	0.961	0.961
TI-9	0.975	0.975	0.972	0.978	0.978	0.975	0.975
TI-10	0.964	0.964	0.959	0.964	0.964	0.964	0.964
TI-11	0.955	0.955	0.949	0.955	0.955	0.955	0.955
TI-12	0.946	0.946	0.939	0.946	0.946	0.946	0.946
TI-13	0.915	0.915	0.908	0.915	0.915	0.915	0.915
TI-14	0.968	0.968	0.962	0.968	0.968	0.968	0.968
TI-15	0.936	0.936	0.930	0.937	0.937	0.936	0.936
TI-16	0.965	0.965	0.961	0.966	0.966	0.965	0.965
TI-17	0.956	0.956	0.952	0.957	0.957	0.956	0.956
TI-18	0.950	0.949	0.945	0.953	0.953	0.950	0.950
TI-19	0.907	0.907	0.897	0.906	0.906	0.907	0.907
TI-20	0.989	0.989	0.988	0.992	0.992	0.989	0.989
TI-21	0.994	0.994	0.990	0.994	0.994	0.994	0.994
TI-22	0.994	0.994	0.991	0.995	0.995	0.994	0.994
TI-23	0.993	0.993	0.989	0.993	0.993	0.993	0.993
TI-24	0.995	0.995	0.992	0.995	0.995	0.995	0.995
TI-25	0.995	0.995	0.992	0.995	0.995	0.995	0.995
TI-26	0.994	0.994	0.992	0.995	0.995	0.994	0.994
TI-27	0.995	0.995	0.992	0.995	0.995	0.995	0.995
TI-28	0.994	0.994	0.992	0.995	0.995	0.994	0.994
TI-29	0.994	0.994	0.991	0.994	0.994	0.994	0.994
TI-30	0.995	0.995	0.992	0.995	0.995	0.995	0.995
TI-31	0.996	0.996	0.994	0.996	0.996	0.996	0.996
TI-32	0.995	0.995	0.992	0.995	0.995	0.995	0.995
TI-33	0.961	0.961	0.955	0.961	0.961	0.961	0.961
TI-34	0.975	0.975	0.969	0.975	0.975	0.975	0.975
TI-35	0.986	0.986	0.982	0.987	0.987	0.986	0.986
TI-36	0.955	0.955	0.948	0.955	0.955	0.955	0.955
TI-37	0.983	0.983	0.980	0.984	0.984	0.983	0.983
TI-38	0.966	0.966	0.960	0.966	0.966	0.966	0.966
TI-39	0.991	0.991	0.988	0.991	0.991	0.991	0.991
TI-40	0.960	0.960	0.955	0.960	0.960	0.960	0.960
TI-41	0.975	0.975	0.969	0.975	0.975	0.975	0.975
TI-42	0.977	0.977	0.970	0.977	0.977	0.977	0.977
TI-43	0.971	0.971	0.965	0.971	0.971	0.971	0.971
TI-44	0.978	0.978	0.972	0.978	0.978	0.978	0.978
TI-45	0.977	0.977	0.972	0.977	0.977	0.977	0.977

Table C.10 MSSIM values for downsizing by factor of 5/6 using various methods.

Test Image	PSNR (dB)						
	prop($\tilde{N}$)			Park [PP06]		Shu [SC07]	
	$\tilde{N} = N$	$\tilde{N} = N_{0.05}$	$\tilde{N} = N_{PT}$	park(I)	park(II)	shu(I)	shu(II)
TI-1	0.941	0.941	0.942	0.942	0.942	0.941	0.941
TI-2	0.978	0.978	0.976	0.980	0.980	0.978	0.978
TI-3	0.962	0.962	0.956	0.963	0.963	0.962	0.962
TI-4	0.954	0.954	0.946	0.954	0.954	0.954	0.954
TI-5	0.988	0.988	0.986	0.989	0.989	0.988	0.988
TI-6	0.982	0.982	0.976	0.982	0.982	0.982	0.982
TI-7	0.987	0.987	0.985	0.989	0.989	0.987	0.987
TI-8	0.968	0.968	0.960	0.968	0.968	0.968	0.968
TI-9	0.979	0.979	0.977	0.981	0.981	0.979	0.979
TI-10	0.971	0.971	0.967	0.972	0.972	0.971	0.971
TI-11	0.963	0.963	0.957	0.963	0.963	0.963	0.963
TI-12	0.958	0.958	0.946	0.958	0.958	0.958	0.958
TI-13	0.930	0.930	0.923	0.930	0.930	0.930	0.930
TI-14	0.974	0.974	0.968	0.974	0.974	0.974	0.974
TI-15	0.947	0.947	0.940	0.948	0.948	0.947	0.947
TI-16	0.972	0.972	0.967	0.973	0.973	0.972	0.972
TI-17	0.963	0.963	0.958	0.963	0.963	0.963	0.963
TI-18	0.957	0.957	0.954	0.960	0.960	0.957	0.957
TI-19	0.924	0.924	0.909	0.925	0.925	0.924	0.924
TI-20	0.990	0.990	0.988	0.991	0.991	0.990	0.990
TI-21	0.995	0.995	0.990	0.995	0.995	0.995	0.995
TI-22	0.995	0.995	0.992	0.995	0.995	0.995	0.995
TI-23	0.994	0.994	0.989	0.994	0.994	0.994	0.994
TI-24	0.996	0.996	0.994	0.996	0.996	0.996	0.996
TI-25	0.996	0.996	0.993	0.996	0.996	0.996	0.996
TI-26	0.995	0.995	0.992	0.995	0.995	0.995	0.995
TI-27	0.996	0.996	0.993	0.996	0.996	0.996	0.996
TI-28	0.995	0.995	0.993	0.996	0.996	0.995	0.995
TI-29	0.995	0.995	0.991	0.995	0.995	0.995	0.995
TI-30	0.996	0.996	0.993	0.996	0.996	0.996	0.996
TI-31	0.997	0.997	0.995	0.997	0.997	0.997	0.997
TI-32	0.996	0.996	0.994	0.996	0.996	0.996	0.996
TI-33	0.970	0.970	0.964	0.970	0.970	0.970	0.970
TI-34	0.980	0.980	0.974	0.981	0.981	0.980	0.980
TI-35	0.989	0.989	0.984	0.990	0.990	0.989	0.989
TI-36	0.965	0.965	0.959	0.966	0.966	0.965	0.965
TI-37	0.987	0.987	0.984	0.988	0.988	0.987	0.987
TI-38	0.973	0.973	0.968	0.974	0.974	0.973	0.973
TI-39	0.992	0.992	0.989	0.993	0.993	0.992	0.992
TI-40	0.969	0.969	0.965	0.969	0.969	0.969	0.969
TI-41	0.981	0.981	0.974	0.981	0.981	0.981	0.981
TI-42	0.982	0.982	0.975	0.982	0.982	0.982	0.982
TI-43	0.977	0.977	0.972	0.978	0.978	0.977	0.977
TI-44	0.982	0.982	0.977	0.983	0.983	0.982	0.982
TI-45	0.982	0.982	0.977	0.983	0.983	0.982	0.982

Table C.11 MSSIM values for downsizing by factor of 6/7 using various methods.

Test Image	PSNR (dB)						
	prop($\tilde{N}$)			Park [PP06]		Shu [SC07]	
	$\tilde{N} = N$	$\tilde{N} = N_{0.05}$	$\tilde{N} = N_{PT}$	park(I)	park(II)	shu(I)	shu(II)
TI-1	0.948	0.949	0.947	0.949	0.949	0.948	0.948
TI-2	0.983	0.982	0.980	0.984	0.984	0.983	0.983
TI-3	0.967	0.967	0.960	0.967	0.967	0.967	0.967
TI-4	0.961	0.960	0.952	0.961	0.961	0.961	0.961
TI-5	0.989	0.989	0.987	0.990	0.990	0.989	0.989
TI-6	0.985	0.985	0.976	0.985	0.985	0.985	0.985
TI-7	0.989	0.989	0.986	0.990	0.990	0.989	0.989
TI-8	0.972	0.972	0.963	0.973	0.973	0.972	0.972
TI-9	0.981	0.981	0.978	0.982	0.982	0.981	0.981
TI-10	0.975	0.975	0.971	0.975	0.975	0.975	0.975
TI-11	0.969	0.969	0.964	0.970	0.970	0.969	0.969
TI-12	0.965	0.964	0.952	0.965	0.965	0.965	0.965
TI-13	0.939	0.938	0.932	0.939	0.939	0.939	0.939
TI-14	0.977	0.977	0.971	0.978	0.978	0.977	0.977
TI-15	0.955	0.954	0.947	0.955	0.955	0.955	0.955
TI-16	0.976	0.976	0.971	0.977	0.977	0.976	0.976
TI-17	0.968	0.968	0.963	0.968	0.968	0.968	0.968
TI-18	0.964	0.964	0.961	0.966	0.966	0.964	0.964
TI-19	0.939	0.938	0.921	0.939	0.939	0.939	0.939
TI-20	0.991	0.991	0.988	0.992	0.992	0.991	0.991
TI-21	0.995	0.995	0.990	0.995	0.995	0.995	0.995
TI-22	0.996	0.996	0.991	0.996	0.996	0.996	0.996
TI-23	0.995	0.994	0.988	0.995	0.995	0.995	0.995
TI-24	0.996	0.996	0.994	0.996	0.996	0.996	0.996
TI-25	0.996	0.996	0.994	0.996	0.996	0.996	0.996
TI-26	0.996	0.996	0.991	0.996	0.996	0.996	0.996
TI-27	0.996	0.996	0.993	0.996	0.996	0.996	0.996
TI-28	0.996	0.996	0.993	0.996	0.996	0.996	0.996
TI-29	0.995	0.995	0.990	0.995	0.995	0.995	0.995
TI-30	0.996	0.996	0.993	0.996	0.996	0.996	0.996
TI-31	0.997	0.997	0.995	0.997	0.997	0.997	0.997
TI-32	0.996	0.996	0.993	0.996	0.996	0.996	0.996
TI-33	0.976	0.976	0.972	0.977	0.977	0.976	0.976
TI-34	0.984	0.984	0.976	0.985	0.985	0.984	0.984
TI-35	0.991	0.991	0.985	0.991	0.991	0.991	0.991
TI-36	0.972	0.972	0.967	0.973	0.973	0.972	0.972
TI-37	0.990	0.990	0.986	0.990	0.990	0.990	0.990
TI-38	0.978	0.978	0.974	0.979	0.979	0.978	0.978
TI-39	0.994	0.994	0.989	0.994	0.994	0.994	0.994
TI-40	0.975	0.974	0.970	0.975	0.975	0.975	0.975
TI-41	0.985	0.984	0.977	0.985	0.985	0.985	0.985
TI-42	0.985	0.984	0.976	0.985	0.985	0.985	0.985
TI-43	0.981	0.981	0.976	0.982	0.982	0.981	0.981
TI-44	0.985	0.985	0.978	0.985	0.985	0.985	0.985
TI-45	0.986	0.986	0.981	0.987	0.987	0.986	0.986

Table C.12 MSSIM values for downsizing by factor of 7/8 using various methods.

Test Image	PSNR (dB)						
	prop($\tilde{N}$)			Park [PP06]		Shu [SC07]	
	$\tilde{N} = N$	$\tilde{N} = N_{0.05}$	$\tilde{N} = N_{PT}$	park(I)	park(II)	shu(I)	shu(II)
TI-1	0.948	0.949	0.947	0.951	0.926	0.948	0.948
TI-2	0.990	0.989	0.986	0.991	0.983	0.990	0.990
TI-3	0.969	0.969	0.960	0.969	0.956	0.969	0.969
TI-4	0.964	0.964	0.954	0.964	0.949	0.964	0.964
TI-5	0.990	0.990	0.987	0.991	0.988	0.990	0.990
TI-6	0.986	0.986	0.978	0.986	0.980	0.986	0.986
TI-7	0.992	0.992	0.988	0.993	0.989	0.992	0.992
TI-8	0.975	0.975	0.962	0.975	0.964	0.975	0.975
TI-9	0.989	0.989	0.984	0.990	0.983	0.989	0.989
TI-10	0.977	0.976	0.972	0.977	0.965	0.977	0.977
TI-11	0.973	0.972	0.967	0.973	0.960	0.973	0.973
TI-12	0.970	0.969	0.956	0.970	0.957	0.970	0.970
TI-13	0.942	0.941	0.935	0.942	0.920	0.942	0.942
TI-14	0.979	0.979	0.969	0.979	0.969	0.979	0.979
TI-15	0.958	0.958	0.950	0.959	0.940	0.958	0.958
TI-16	0.975	0.975	0.967	0.975	0.965	0.975	0.975
TI-17	0.970	0.970	0.964	0.970	0.955	0.970	0.970
TI-18	0.973	0.972	0.968	0.974	0.960	0.973	0.973
TI-19	0.950	0.949	0.934	0.951	0.930	0.950	0.950
TI-20	0.992	0.992	0.989	0.992	0.990	0.992	0.992
TI-21	0.995	0.995	0.992	0.996	0.994	0.995	0.995
TI-22	0.996	0.996	0.992	0.996	0.994	0.996	0.996
TI-23	0.995	0.995	0.991	0.995	0.992	0.995	0.995
TI-24	0.996	0.996	0.993	0.997	0.994	0.996	0.996
TI-25	0.996	0.996	0.994	0.997	0.994	0.996	0.996
TI-26	0.996	0.996	0.993	0.996	0.994	0.996	0.996
TI-27	0.996	0.996	0.993	0.997	0.994	0.996	0.996
TI-28	0.996	0.996	0.993	0.996	0.993	0.996	0.996
TI-29	0.995	0.995	0.992	0.996	0.993	0.995	0.995
TI-30	0.996	0.996	0.993	0.997	0.994	0.996	0.996
TI-31	0.997	0.997	0.994	0.998	0.995	0.997	0.997
TI-32	0.996	0.996	0.993	0.996	0.993	0.996	0.996
TI-33	0.981	0.980	0.975	0.981	0.969	0.981	0.981
TI-34	0.986	0.986	0.978	0.987	0.979	0.986	0.986
TI-35	0.992	0.992	0.987	0.992	0.987	0.992	0.992
TI-36	0.977	0.977	0.971	0.977	0.963	0.977	0.977
TI-37	0.991	0.991	0.987	0.991	0.983	0.991	0.991
TI-38	0.981	0.981	0.976	0.982	0.972	0.981	0.981
TI-39	0.994	0.994	0.990	0.994	0.990	0.994	0.994
TI-40	0.977	0.977	0.972	0.977	0.964	0.977	0.977
TI-41	0.987	0.987	0.979	0.988	0.980	0.987	0.987
TI-42	0.987	0.986	0.981	0.987	0.979	0.987	0.987
TI-43	0.983	0.983	0.977	0.984	0.973	0.983	0.983
TI-44	0.987	0.987	0.979	0.987	0.981	0.987	0.987
TI-45	0.988	0.988	0.981	0.988	0.980	0.988	0.988

Fig. C.1 Average MSSIM difference for various downsizing factors for $Q = 2$ to 7 in (a) to (e), respectively.

Fig. C.2 Average MSSIM difference for various downsizing factors for $Q = 8$ to 12 in (a) to (e), respectively.