

Multi-Scale Edge-Guided Image Gap Restoration

A thesis submitted for the degree of Doctor of Philosophy

By

Bahareh Langari

School of Engineering & Design

Brunel University

February 2016

Abstract

The focus of this research work is the estimation of gaps (missing blocks) in digital images. To progress the research two main issues were identified as (1) the appropriate domains for image gap restoration and (2) the methodologies for gap interpolation.

Multi-scale transforms provide an appropriate framework for gap restoration. The main advantages are transformations into a set of frequency and scales and the ability to progressively reduce the size of the gap to one sample wide at the transform apex. Two types of multi-scale transform were considered for comparative evaluation; 2-dimensional (2D) discrete cosines (DCT) pyramid and 2D discrete wavelets (DWT).

For image gap estimation, a family of conventional weighted interpolators and directional edge-guided interpolators are developed and evaluated. Two types of edges were considered; ‘local’ edges or textures and ‘global’ edges such as the boundaries between objects or within/across patterns in the image.

For local edge, or texture, modelling a number of methods were explored which aim to reconstruct a set of gradients across the restored gap as those computed from the known neighbourhood. These differential gradients are estimated along the geometrical vertical, horizontal and cross directions for each pixel of the gap.

The edge-guided interpolators aim to operate on distinct regions confined within edge lines. For global edge-guided interpolation, two main methods explored are Sobel and Canny detectors. The latter provides improved edge detection.

The combination and integration of different multi-scale domains, local edge interpolators, global edge-guided interpolators and iterative estimation of edges provided a variety of configurations that were comparatively explored and evaluated.

For evaluation a set of images commonly used in the literature work were employed together with simulated regular and random image gaps at a variety of loss rate. The performance measures used are the peak signal to noise ratio (PSNR) and structure

similarity index (SSIM). The results obtained are better than the state of the art reported in the literature.

Statement of copyright

The copyright of this thesis rests with the author, Bahareh Langari. No parts from it should be published without his prior written consent and the information derived from it should be acknowledged.

Declaration

The work described in this thesis has not been previously submitted for a degree in this or any other university, and unless otherwise referenced it is the author's own work.

Table of Contents

<i>Abstract</i>	ii
<i>Statement of copyright</i>	iv
<i>Declaration</i>	v
<i>List of Abbreviations</i>	xiv
<i>List of Symbols</i>	xv
<i>Acknowledgments</i>	xviii
<i>Publications</i>	xix
1. Introduction.....	2
1.1 A Brief History of Image and Visual Communication	3
1.2 Thesis Subject: Image Gap Restoration	5
1.2.1 Causes of Image Gaps.....	6
1.2.2 Different Approaches to Recovery/Restoration of Images with Lost Segments....	6
1.2.3 Standard Test Images and Restoration Performance Measures	7
1.2.4 Image Processing Challenges in Image Gap Restoration.....	7
1.2.5 List of Contributions to Image Restoration.....	8
1.3 Organisation of the thesis.....	9
2. Literature Review.....	13
2.1 Introduction.....	13
2.2 The Packet Loss Recovery and Concealment Methods	14
2.2.1 Automatic Request for Retransmission (ARQ).....	15
2.2.2 Forward Error Correction (FEC).....	17
2.2.3 Error Concealment (EC) Method.....	19
2.3 Image Transformation Methods for Gap Interpolation.....	21
2.3.1 Discrete Cosine Transform (DCT).....	21
2.3.2 Discrete Wavelet Transform (DWT)	24
2.3.3 Pyramid Transforms.....	26
2.4 Interpolation of Missing Gaps.....	29
2.4.1 Non-directional Interpolation Methods.....	29
2.4.2 Directional Interpolation Methods	33
2.4.3 Combined Directional and Non-Directional Interpolation Methods.....	42
2.5 Conclusion	43

3.	Image Processing Methods and Tools	46
3.1	Introduction.....	46
3.2	Image Processing History	46
3.3	Image Processing Basics.....	47
3.3.1	Image Acquisition.....	47
3.3.2	Image Sampling and Quantisation	48
3.3.3	Digital Image Representation.....	49
3.3.4	Image File Formats and Image Compression Algorithms	51
3.3.5	Packet loss.....	55
3.4	Discrete Cosine Transform (DCT).....	55
3.4.1	DCT Equations and Basis Functions	58
3.4.2	DCT for Resizing Images	62
3.4.3	Zig-zag Scanning	63
3.5	Discrete Wavelet Transforms.....	64
3.5.1	Wavelet Analysis-Synthesis.....	66
3.5.2	Components of Wavelet Decomposition: Approximation and Details.....	69
3.6	Edge Detection.....	71
3.6.1	Sobel Edge Detection.....	76
3.6.2	Canny Edge Detection	77
4.	Image Gap Restoration Using a Multi-scale DCT Pyramid.....	82
4.1	Introduction.....	82
4.2	Multi-Scale Discrete Cosine Transform Pyramid	83
4.3	Gap Restoration using Multi-scale 2D-DCT.....	89
4.4	Alternative methods of Interpolation	90
4.4.1	Frequency Transform, DCT Based, Interpolation.....	93
4.4.2	Nearest Neighbour Interpolation.....	95
4.4.3	Linear Interpolation.....	98
4.4.4	Cubic Interpolation	100
4.4.5	Spline Interpolation.....	103
4.4.6	Comparison Between Different Interpolation Methods	106
4.5	Evaluation for Regular Loss Pattern	107
4.6	Conclusion	112
5.	Multi-scale Edge-Guided DCT Image Gap Restoration	115
5.1	Introduction.....	115

5.2 Local Edges, Textures in Multi-Scale Image Gap Interpolation.....	116
5.2.1 Local Interpolation of a Single Missing Pixel at the Apex of a Multi-Scale Pyramid.....	116
5.2.2 Local Block Interpolation at $[N \times N]$ Pyramid Scale	118
5.2.3 Experimental Result for Local Edge-Guided Interpolation	122
5.3 Global Edge-Guided Image Gap Interpolation	132
5.3.1 Edge Detection.....	134
5.3.2 Experimental Results	140
5.4 Conclusion	154
6. Multi-scale Edge-Guided Wavelet Image Gap Restoration.....	157
6.1 Introduction.....	157
6.2 Wavelet Analysis-Synthesis of Distorted Images	158
6.2.1 Components of Wavelet Decomposition; Approximation and Details.....	158
6.2.2 Types of Wavelength Applicable to the Gap Restoration Problem	160
6.3 Wavelet Reconstruction of Distorted Images – Conventional vs Edge-Guided Interpolation.....	167
6.3.1 Conventional (non-edge-based) Interpolation.....	167
6.3.2 Edge-Guided Interpolation.....	167
6.4 Evaluation for Regular Loss Pattern	176
6.4.1 Regular Loss Pattern 8×8 Missing Block	176
6.4.2 Regular Loss Pattern 16×16 Missing Block.....	180
6.5 Evaluation for Random Loss Pattern	182
6.6 Run-Time Comparison.....	187
6.7 Conclusion	188
7. Conclusion: Discussion and Future Works	191
7.1 Conclusion	191
References.....	203

Table of Figures

Figure 1.1: Boulevard du Temple - Paris, by Daguerre, 1838, includes the earliest known candid photograph of a person – man having his boots polished.	4
Figure 2.1: The original image of peppers and right, the image with 40% random missing macro blocks (8×8).	14

Figure 2.2: Framework of the automatic request repeat (ARQ) process, from the top row: with no error, with errors and retransmission.	16
Figure 2.3: Diagram of the forward error correction (FEC) method (\times show lost bits).....	18
Figure 2.4: a) The missing pixel is interpolated by the values of four pixels on all the boundaries. b) The missing pixel is estimated by two pixels on the nearest two boundaries.	23
Figure 2.5: Recovered Lena image by Bajic´ & Woods (2003) with 25% packet loss.....	25
Figure 2.6: Illustration of DCT-Pyramid processing. Three levels of a two-dimensional DCT pyramid coding and reconstruction IDCT pyramid (Tan & Ghanbari, 1992).	28
Figure 2.7: Weighting average interpolation, one missing MB (8 \times 8), four available surrounding blocks.....	30
Figure 2.8: Diagram of a lost block, range block, domain block and searching range block.	33
Figure 2.9: Missing MB (8 \times 8) with eight available surrounding MBs (8 \times 8).....	35
Figure 2.10: Eight edge directions.	35
Figure 2.11: POCS iterative process on a missing block and eight available surrounding MBs (Sun & Kwok, 1995).	36
Figure 2.12: Multi-directional recursive nonlinear filtering (MRNF) and eight interpolation directions.....	37
Figure 2.13: Corrupted Lena, the restored Lena by (NURBS) interpolation and the zoomed imperfections.....	38
Figure 2.14: Restoration by dominant SDVs: DSDV1 and DSDV2.	40
Figure 2.15: Comparison of different EC algorithms on Lena with 25% loss (8 \times 8 isolated block loss). (a) Lossy image, (b) Bayesian EC (Zhai et al., 2010), (c) BBF (Prochazka, Vysata & Jerhotova, 2010), (d) Sequential EC (Li & Orchard, 2002), (e) FSE(Meisinger & Kaup, 2004) and (f) POCS-based EC (Park <i>et al.</i> , 2005).	41
Figure 2.16: Framework of content adaptive spatial error concealment.	42
Figure 3.1: Input-output block diagram of an imaging sensor. Input: light energy (photons), output: sampled and digitised voltage.....	48
Figure 3.2: a) continuous image captured by the imaging machine, b) sampling and quantisation of the captured image.	49
Figure 3.3: a) sensor array with a continuous image, b) sampled and quantized image.	49
Figure 3.4: A part of Lena image showing the image pixels. Matrix of 8 \times 8 pixel values which is marked by a red square in the image.	50
Figure 3.5: Lena.BMP 512 \times 512 pixels, 768 KB and the individual RGB components.	50
Figure 3.6: Compression framework of both the lossy and lossless methods.....	52
Figure 3.7: Lena.JPEG 512 \times 512 pixels, 61 KB and its individual RGB components Lena.JEPG.	54
Figure 3.8: Wavelet-based compression method.	54
Figure 3.9: a) Original Lena image, b) Lena image with a 25% regular 8 \times 8 missing macro block and c) Lena image with a 10% random 8 \times 8 missing macro block.	55
Figure 3.10: Original colour Lena and the energy compression map of this image after applying a large 512 \times 512 DCT.....	56
Figure 3.11: Illustration of 64 basis functions of 8 \times 8 2D DCT. Top left corner shows lowest (zero) frequency and bottom right corner the highest. Note, energy colour coded: white highest energy and black zero energy.	57
Figure 3.12: A image processing flow chart incorporating DCT as the main transform for data representation and data reduction.....	59

Figure 3.13: From left to right, DCT of Lena image and sub-images of size: (512 × 512), (256 × 256), (128 × 128) and (64 × 64) pixels, respectively.	60
Figure 3.14: Left, original Boat image and right, the reconstructed image using only 7% of the low-frequency 2D-DCT coefficients and applying 2D-IDCT (8×8 macro blocks).	61
Figure 3.15: Image size expansion: zero-padding in the transform domain to oversample or reconstruct a truncated input image.	63
Figure 3.16: Actual application to images 8×8 MBs (512×512) transformed to 16×16 (1024×1024) or 32×32 (2048×2048); useful for fitting to screen displays of different sizes.	64
Figure 3.17: Zig-zag scanning for an MB (8×8) in 2D-DCT.	64
Figure 3.18: Schematic diagram of a 1D wavelet transform.	66
Figure 3.19: Schematic diagram of sub-band decomposition for one level of a 2D wavelet transform.	67
Figure 3.20: (a) Wavelet pyramid image decomposition coefficients in three- stages; and (b) its application to Lena image of size 512×512.	70
Figure 3.21: Block diagram of the three-stage wavelet pyramid image decomposition and its application to the Lena image.	71
Figure 3.22: Experiments using different edge detection methods on Lena (512×512); note Canny has two threshold parameters in comparison to one threshold parameter for other methods.	75
Figure 3.23: Application of the Sobel filter: a) original Lena (512×512), b) x direction (G_x), c) y direction (G_y), and d) the combined gradient direction.	77
Figure 3.24: Application of the Sobel filter to zoomed Lena (512×512) in x , y and the combined gradient directions.	77
Figure 3.25: Diagram of the Canny edge detector.	79
Figure 4.1: Multi-scale pyramid decomposition.	83
Figure 4.2: Four layer pyramid decomposition using 8×8 macro blocks.	84
Figure 4.3: Five layer pyramid decomposition using 16×16 macro blocks.	84
Figure 4.4: Block diagram of the three-stage DCT pyramid image decomposition and its application to the Foreman image for a missing block size of 8×8.	86
Figure 4.5: Block diagram of the four-stage DCT pyramid image decomposition and its application to the Peppers image for a missing block size of 16×16.	87
Figure 4.6: Reconstruction from individual pyramid layers: layer 1 (down-sampled by 2×2), layer 2 (down-sampled by 4×4) and layer 3 (down-sampled by 8×8).	88
Figure 4.7: Block diagram of image gap restoration using a multi-scale DCT pyramid.	91
Figure 4.8: Up-sampling a 2×2 macro block into a 4×4 pixel grid by 2D interpolation (spline).	91
Figure 4.9: DCT based interpolation for up-sampling by factor of 2.	93
Figure 4.10: Known and missing pixels sizes (1×1) in the apex layer and (2×2) in the next layer down.	94
Figure 4.11: Diagram of the nearest neighbour interpolation algorithm.	95
Figure 4.12: Up-sampling a 2×2 macro block into a 4×4 pixel grid by NN interpolation.	96
Figure 4.13: Illustration of a one-dimensional nearest neighbour (red) interpolated through a number of known data samples (blue).	97
Figure 4.14: Illustration of up-sampling of a 2 dimensional mesh-grid (blue) by a factor of 2 using a nearest neighbour 2D interpolator (red).	97

Figure 4.15: Diagram of the linear interpolation algorithm.	98
Figure 4.16: Illustration of a one-dimensional linear curve (red) interpolated through a number of known data samples (blue).	99
Figure 4.17: Illustration of up-sampling of a 2 dimensional mesh-grid (blue) by a factor of 2 using a linear 2D interpolator (red).	99
Figure 4.18: Diagram of a cubic interpolation algorithm.	101
Figure 4.19: Illustration of a one-dimensional cubic curve (red) interpolated through a number of known data samples (blue).	102
Figure 4.20: Illustration of up-sampling of a 2 dimensional mesh-grid (blue) by a factor of 2 using a cubic 2D interpolator (red).	102
Figure 4.21: Illustration of a one-dimensional spline curve (red) interpolated through a number of known data samples (blue).	105
Figure 4.22: Illustration of up-sampling of a 2 dimensional mesh-grid (blue) by a factor of 2 using a B-spline 2D interpolator (red).	105
Figure 4.23: Illustration of a one-dimensional NN curve (green), linear curve (red), cubic curve (blue) and spline curve (yellow) interpolated through a number of known samples (blue).	107
Figure 4.24: From left to right, original images, images with 25% MB (size = 8×8) loss rate, restored and zoomed images on Lena, Peppers, Man and Boat for spline interpolation.	111
Figure 5.1: Directional interpolation for each missing pixel at the apex of a multi-scale pyramid (in eight possible directions).	117
Figure 5.2: Diagram of the process in the local edge-guided interpolation.	118
Figure 5.3: Interpolation using the neighbourhood edges, inferred from the pixels available (on three sides of the missing pixel) in two directions: horizontal (H) (red arrow) and vertical (V) (green arrow).	119
Figure 5.4: The interpolation inferred from four possible directions (represented by four different colours) combined with the pixel estimate (white dot).	119
Figure 5.5: Illustration of pyramid DCT decomposition of Lena. A 8×8 gap at the base is transformed to a single missing pixel at the apex and directional interpolation composed of local edge information is used to infer missing pixel values. After up-sampling the estimates are used to replace the gap at a lower pyramid layer and the process is continued to the pyramid base.	120
Figure 5.6: Post-processing: Blending of block estimates with surrounding pixels for a missing block of size 2×2.	121
Figure 5.7: From left to right; the original images, the image with 25% MB loss (8×8 isolated block loss) and the restored images for Lena, Pepper, Man and Foreman. ...	126
Figure 5.8: From left to right: the original Lena, Peppers, Man and Foreman images, the images with 10% random missing MBs and the restored image for local edge interpolation.	129
Figure 5.9: From left to right: the original images, corrupted images (16×16 isolated block loss), the restored and zoomed in images for Lena, Peppers, Man and Foreman for local edge-guided interpolation.	130
Figure 5.10: Three-stage DCT pyramid image decomposition (D-Interp = Directional Interpolation).	133
Figure 5.11: Segmentation of pixels with a global edge detector.	134

Figure 5.12: From left to right an estimated image of Lena after local edge-guided interpolation in the last stage, and edge-based segmentation of an image with the Sobel edge detector.....	134
Figure 5.13: Application of the proposed multi-scale method to restoration of a corrupted image of Lena with 25% of 8×8 blocks loss with the Sobel filter.	135
Figure 5.14: Missing pixel (red pixel) might lie in 4 different possible edge directions (white).....	136
Figure 5.15: Areas next to the edges (white), the missing pixels could be any of the black pixels.....	136
Figure 5.16: Areas next to the edges (white), in the horizontal, left cross, right cross and vertical directions.....	137
Figure 5.17: Application of the Sobel filter to Lena in x , y and the combined gradient directions.....	137
Figure 5.18: Thresholding the output of the Sobel filter, showing input G and output G_{thresh}	138
Figure 5.19: Canny edge detector for multi-scale Lena and Peppers at scales, from left to right: (a) 512^2 , (b) 256^2 , (c) 128^2 , (d) 64^2	139
Figure 5.20: Three stage of iteration for Lena and theirs PSNRs.	140
Figure 5.21: From left to right, the original images, the image with 25% missing MBs and the restored images for Lena, Peppers, Man and Foreman.	144
Figure 5.22: From left to right, the original images, corrupted images (16×16 isolated block loss), the restored images and zoomed in images for Lena, Peppers, Man and Foreman (Canny case).....	148
Figure 5.23: From left to right; the original Lena, Peppers, Man and Foreman images, the image with 10% random missing MBs and the restored image.....	150
Figure 5.24: From left to right; the original Peppers image, the image with 40% random missing MBs and the restored image.	151
Figure 5.25: Top: from left to right, the original Lena and the image with 25% regular missing MBs (8×8). Bottom: from left to right, the reconstructed image with local edge-guided interpolation and global edge-guided interpolation.....	151
Figure 5.26: From left to right, the original image, corrupted image (8×8 isolated block loss) and the restored image for my own picture, PSNR at 36.60 dB.	152
Figure 5.27: From left to right, the original image, corrupted image (16×16 isolated block loss) and the restored image for Dolat-Abad Garden, Yazd, Iran.....	152
Figure 5.28: Experiment on a lost block size of 8×8 pixels of the “Man” image. (a) Original image 512×512, (b) damaged image of one missing block out of every four. Restoration using the methods of (c) Ancis and Giusto (PSNR = 25:47 dB), (d) Sun and Kwok (PSNR = 27:25 dB), (e) Hemami and Meng (PSNR = 27:65 dB), (f) Shirani <i>et al.</i> (PSNR = 27:44 dB), (g) Alkachouh and Bellanger (PSNR = 27:94 dB), (h) Park <i>et al.</i> (PSNR = 29:87 dB), (i) proposed method (31.56 dB).....	153
Figure 6.1: Components of wavelet decomposition; approximation and details for lossy Lena (block loss size = 8×8), in 256×256, 128×128 and 64×64.....	159
Figure 6.2: Performance comparisons using nine different wavelet types for the whole distorted Lena (macro block loss size = 8×8).....	162

Figure 6.3: Performance comparisons using nine different wavelet types for the macro block segmented distorted Lena (macro block loss size = 8×8), distortion pattern = regular, distortion rate = 25%.	164
Figure 6.4: The scaling function and wavelet for Daubechies' wavelet (db1).	166
Figure 6.5: Impulse responses (left) and frequency responses (right) of the decomposition and reconstruction filters for the db1 bi-orthogonal wavelet.	166
Figure 6.6: Diagram of the process of local-edge guided wavelet interpolation.	168
Figure 6.7: From left to right, the original image, the image with 25% MB loss (8×8 isolated block loss) and the restored image for Lena with local-edge guided wavelet interpolation.	169
Figure 6.8: Diagram showing the process of local, global edge-guided interpolation for the coefficients parts, which is followed by the addition of the restored details part (D-Interp = Directional Interpolation).	170
Figure 6.9: From left to right, the original Lena, Lena with 25% macro block loss (8×8) and restored Lena with global edge guided interpolation.	171
Figure 6.10: Canny edge detector for a multi-scale wavelet on Lena at scales, from left to right: (a) 512^2 ; (b) 256^2 ; (c) 128^2 ; and (d) 64^2 .	171
Figure 6.11: Block diagram of the three-stage wavelet pyramid image decomposition on the Lena image for missing block size of 8×8 .	172
Figure 6.12: (a) affected coefficients matrices, cV, cH, and cD by missing pixels (8×8); and (b) the restored cV, cH and cD on Lena.	173
Figure 6.13: Performance comparisons with twenty iterations for a macro block loss rate of 25% (8×8) on Lena for the last stage of reconstruction of the cA coefficient part.	173
Figure 6.14: Four stages of iteration for Lena starting from the left (first stage) and finishing on the right (fourth stage).	174
Figure 6.15: (a) original graphs of cA, cH, cV and cD; and (b) graphs of cA, cH, cV and cD after thresholding.	175
Figure 6.16: (a) original Lena; (b) Lena with 25% macro block loss (8×8); (c) Lena with macro block loss (16×16); and (d) Lena with 10% random MB loss (8×8).	175
Figure 6.17: From left to right, the original images, the image with 25% missing MBs and the restored images for Lena, Peppers, Man and Foreman.	179
Figure 6.18: From left to right, the original images, corrupted images (16×16 isolated block loss), the restored images and zoomed in images for Lena, Peppers, Man and Foreman.	181
Figure 6.19: a) Original Lena image (512×512), b) Corrupted image (16×16 block loss). Images restored using the methods of (c) Sun & Kwok (PSNR = 34.95 dB), (d) Li & Orchard (PSNR = 37.41), (e) Kim, Koo & Jeong (PSNR = 37.37 dB) and (f) the proposed method (PSNR = 37.65 dB).	182
Figure 6.20: Top from left to right; the original Peppers, the image with 25% random missing MBs and the restored image and bottom the original Peppers, the image with 40% random missing MBs and the restored image.	184
Figure 6.21: From left to right; the original Lena, Peppers, Man and Foreman images, with 10% random missing MBs and the restored images.	185
Figure 6.22: Experiment on a lost block size of 8×8 pixels of the "Man" image. (a) Original 512×512 and (b) damaged image of one missing block out of every four. Restoration using the methods of (c) Ancis and Giusto (PSNR = 25:47 dB), (d) Sun and Kwok (PSNR = 27:25 dB), (e) Hemami and Meng (PSNR = 27:65 dB), (f) Shirani <i>et al.</i> (PSNR = 27:44 dB), (g)	

Alkachouh and Bellanger (PSNR = 27:94 dB), (h) Park *et al.* (PSNR = 29:87 dB) and (i) the proposed method (31.61 dB)..... 186
 Figure 6.23: From left to right, original image, corrupted image (8×8 isolated block loss) and the restored image for the Dolat-Abad Garden, Yazd, Iran. 187

List of Abbreviations

BMP Bitmap
 C Cross direction
 CAT (CT for short) Computerised axial tomography
 DB1 Daubechies version one
 1-D One dimensional
 2-D Two dimensional
 DCT Discrete cosine transform
 DFT Discrete Fourier transform
 DWT Discrete wavelet transform
 EC Error concealment
 EMI Electromagnetic interference
 FEC Forward error correction
 FT Fourier transform
 GDI Graphics device interface
 H Horizontal direction
 HPF High-pass filter
 IP Internet protocol
 ITU International Telecommunications Union
 JPEG Joint Photographic Expert Group
 JPEG2000 Joint Photographic Experts Group in 2000
 KL Karhunen–Loève (Transform)
 LC Left cross
 LP Linear prediction
 LPF Low-pass filter
 MB Macro block
 ML Maximum likelihood
 MMSE Minimum mean square error
 MSE Mean squared error

NACK Negative acknowledgment
OS Operating system
Pixel Picture element
PC Principal component
PDF Probability distribution function
PNG Portable network graphics
PLC Packet loss concealment
PSNR Peak signal to noise ratio
RC Right cross
RGB Red, green and blue
RI Region of interest
RMS Root mean square
SNR Signal-to-noise ratio
SSIM The structural similarity index
TIFF Tagged image file format
UDP User datagram protocol
V Vertical

List of Symbols

A matrix of input vector

A_1 and A_2 are matrices derived from the 2-D DCT matrix

$A_{m,n}$ missing sample in the (m,n) directions

A_M matrix of sizes M

A_N matrix of sizes N

$A(x, y)$ intensity values (an amplitude) of the pixel (A) in x and y directions

$c(i, j)$ the closest point of the input samples to the unknown pixel p

d_x represents a small change in distance in the x direction

d_y represents a small change in distance in the y direction

$\frac{dy}{dx}$ = edge

F_s Frequency

$F_{(p,q)}$ the DCT coefficient in row k_1 and column k_2 of the DCT matrix. Coordinates in the frequency domain

$f(i,j)$ is the intensity of the pixel in row i and column j ;

$f_{\sigma}(x, y)$ denotes the 2D optimal filter
 $f_{\sigma}()$ 1D Gaussian's derivative
 G_x gradient in x direction
 G_y gradient in y direction
 $g[n]$ high-pass filter
 $G_{\sigma}()$ denote the 1D Gaussian function
H horizontal direction
 $h[n]$ low-pass filter
K the element to change the filter into different edge detectors
 $L(x, y)$ The Laplacian of an image
M M data samples
M-by-N-by-3 array of RGB values. The first, second and third matrix contains the red, the green and the blue components
 $M \times N$ matrix by M rows and N columns
N signal samples
2N spectral *samples*
p entry of the DCT of an image
q entry of the DCT of an image
s the pyramid layer
 S_x differential operator in the x direction
 S_y differential operator in the y direction
T The DCT basis functions
 T' Inverse of matrix T
 T_{pq} DCT Matrix with p and q entries
V vertical direction
x Signal
x x direction
X to vectors from $M_s \times N_s$ input pixels
 $x[n]$ original signal
y y direction
Y $M_s \times N_s$ DCT output
 ∞ infinite
 $\theta(x, y)$ represents the direction of the gradient
 ψ The mother wavelet function
 $\emptyset(x)$ 1D wavelets

$\phi(y)$ 1D wavelets
 ϕ the scaling function
 σ variance

Acknowledgments

I would like to express my special appreciation to my first supervisor Prof. John Stonham for his help and support during my course. His experience and appreciation were invaluable. I would like to thank Dr. Ali Mousavi for his guidance throughout the course of this work.

I would like to thank Prof. Saeed Vaseghi for his assistance and support continuously throughout the course of my PhD research.

I would like to extend my sincere gratitude to my parents and family for all their love, support and encouragement. I am also thankful to my fellow researchers at Brunel University.

I owe the most to my loving husband, Farzad, for his companionship, selfless support and love.

Bahareh Langari

Publications

- **Bahareh Langari**, “Strategies for Packet Loss Concealment for Multimedia,” *Poster Presentation*, School of Engineering and Design, *RESCON*, Brunel University, 2012.
- **Bahareh Langari**, “Multiresolution Edge-Guided Gap Restoration,” School of Engineering and Design, *RESCON*, Brunel University, 2013.
- Seyed Kamran Pedram, Saeed Vaseghi and **Bahareh Langari**, “Audio packet loss concealment using motion-compensated spectral extrapolation,” *IEEE ISSPIT*, pp: 434-439, 2013.
- **Bahareh Langari**, Saeed Vaseghi and Seyed Kamran Pedram, “Multi-resolution edge-guided image gap restoration,” *IEEE ISSPIT*, pp: 374-379, 2013.
- Seyed Kamran Pedram, Saeed Vaseghi and **Bahareh Langari**, “Audio packet loss concealment using spectral motion”, *IEEE ICASSP*, pp: 6707-6710, 2014.
- **Bahareh Langari**, “A Novel Edge-Guided Interpolation for Image Gap Restoration,” School of Engineering and Design, *RESCON*, Brunel University, 2014.
- **Bahareh Langari**, Saeed Vaseghi and Ales Prochazka, “Edge-Guided Image Gap Interpolation Using Multi-scale Transformation,” Accepted to *IEEE Transaction on Image Processing* 2015.
- **Bahareh Langari**, John Stonham and Ali Mousavi, “Image Loss Concealment Using Edge-Guided Interpolation And Multi-Scale Transformation,” *IEEE SP/SPE*, pp: 391-396, Utah, USA, 2015.

Chapter 1

Introduction

Chapter 1

1. Introduction

The subject of this thesis is the development of image processing models for image gap restoration. The gaps or missing data can be due to loss in transmission or damage to the storage medium. The restoration methods strive to make an optimal numerical estimate of the missing parts of an image consistent with the available data.

The estimation of gaps in data is a fundamental problem in classical estimation and interpolation theory with numerous applications in the fields of numerical data analysis, including econometrics, scientific data analysis and communication signal processing (Gonzales and Woods, 2008).

The interpolation problem is particularly challenging when applied to the restoration of gaps in images due to the non-homogenous, non-stationary nature of image content that is often composed of a mixture of different objects, textures, patterns and edges on the various sides of the gaps. The interpolation should be capable of restoring the gap in a way that is consistent with the non-stationary patterns surrounding it.

For a stationary process, that is, one without underlying space-time variations or edges, a low-pass filter is the ideal interpolator as it can provide a smooth replacement of the missing gap signal. However a typical image is a highly non-stationary, space-variant process and hence, a low-pass filter/smoother will not provide satisfactory results for the regions of image gaps that span edges and different patterns on the various sides of the gaps. The low-pass interpolation result is likely to be a blurred smeared mix of different patterns that surround any gap.

Consequently, for satisfactory interpolation of gaps in images, advanced image processing methods need to be developed that involve combining appropriate signal

analysis and transform tools so as to take account of edges and patterns as well as deal with gap sizes that are beyond the capability of a simple ‘ideal’ low-pass filter interpolator.

The application of image gap restoration has a multitude of applications, such as in the medical, nuclear physics, astrophysics and image/video communication field. Image gap loss concealment is particularly important and is the subject of this research thesis.

1.1 A Brief History of Image and Visual Communication

Visual records are the most information-rich, and valuable form of communication with a long history taking various forms such as engravings, sculptures, paintings, and in modern times electronic-media image sequences.

The recording of historical/social/artistic events or scientific observations by means of images begins with the appearance of visual communication systems in the Neolithic era, as evident from archaeological records of different historical periods (Bonatz, 2007), and has continued up until the recent space images of Pluto transmitted by the New Horizons spacecraft at the time of writing this thesis.

The idea of modern photography is attributed to Thomas Wedgwood some time during the 18th century. He appears to have been the first person, reliably documented, to have used light-sensitive chemicals to capture silhouette images on durable material and the first to have attempted to photograph the image formed in a camera obscura; a device that consists of a box or room with a hole in one side (Katzman, n.d.). The date of his first experiments in photography is unknown, but he is believed to have indirectly advised James Watt on the practical details prior to 1800 (Greenspun, 1999).

The process of photography was further developed by Nicéphore Niépce and Louis Daguerre, (Greenspun, 1999). Nicéphore Niépce started his work on light sensitive varnished materials used in lithography in 1814. In order to create a light sensitive plate, he covered a coated pewter plate with asphaltum and then covered all of this in wax before placing it in sunlight. To complete the process, the plate was washed

with a solvent and etched in an acid bath to remove the asphaltum and the result was a printed plate. Later, the same method was utilised for the photography of the first camera image (Katzman, n.d.). However, the technique was time consuming and the result was not clear. Louis Daguerre continued the experiment and proposed a new silver-plated copper technique in 1840, which resulted in mirror-like images (Yaroslavsky, 2011). The first captured image of a person is demonstrated in Figure 1.1, which is a view from Louis Daguerre's window. It took around ten minutes of exposure, hence the traffic did not appear in the image.

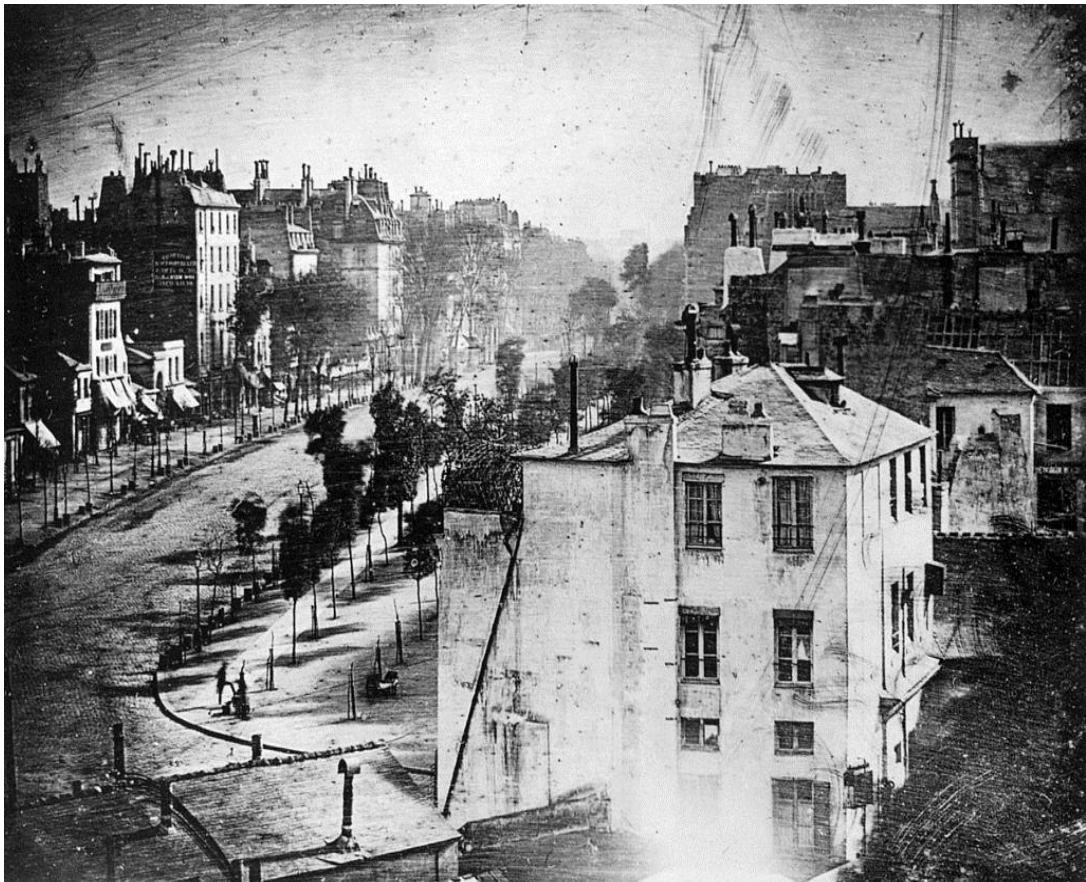


Figure 1.1: Boulevard du Temple - Paris, by Daguerre, 1838, includes the earliest known candid photograph of a person – man having his boots polished.

From then on, the hardware and methods of capturing and processing image photography improved dramatically and nowadays, digital image recording and processing is commonplace with the advent of high resolution stand-alone and integrated cameras. The most recent technology integrated in mobile phones offers as much as a massive 41 mega pixels resolution (with an uncompressed file size of 123 megabytes), which for a normal size image, surpasses natural human eye

resolution (Microsoft, 2016). State of the art digital images are very large multidimensional files for which advanced image processing and compression methods have been developed.

1.2 Thesis Subject: Image Gap Restoration

The purpose of image restoration is to recover the best approximation of an original image degraded by gaps, noise or through blurring (Kim, 2002).

The focus of this work is on the development of advanced image processing methods for image gap restoration. Image restoration has a history that goes back to the scientific endeavour of the space programmes of both the United States and the former Soviet Union in the 1950s. Subnormal environment conditions degraded the quality of pictures obtained from most space missions. Owing to the amount of resources spent on space projects and avid public and media interest, there was a need for using the available information to perform image enhancement and restoration (Banham and Katsaggelos, 1997). The advent of digital computers and digital image processing in 1960s led to the development of digital transformations, estimation theory and algorithms for a host of image processing applications including image restoration.

Applications of image restoration include scientific explorations, medical imaging, legal/forensic investigation, reissuing films or using clips of old films in new productions, image/video coding and decoding, consumer photographs, image reconstruction in astronomy, radar imaging, tomography and image reconstruction for damaged historical and art objects. The wide range of applications demonstrates that there is an important role for effective image restoration methods in today's world.

The focus in this thesis is on the restoration of still images, which presents major research challenges that require in-depth study of the main issues and the current methods, development of new improved methods and solutions and comparative evaluations of the proposed algorithms. Note that the methods developed for still

images can also be applied to video, although the use of temporal information across frames would be expected to result in further improvements.

1.2.1 Causes of Image Gaps

The cause of loss of data and gaps in images can be due one of the following (Ira, 2008):

- 1) Physical degradation or scratches of storage material, this is particularly the case with older archived image materials;
- 2) Losses due to electromagnetic interference (EMI) during recording or transmission of signals;
- 3) Losses due to transmission signal outage, particularly in wireless, that is errors in synchronisation or equalisation;
- 4) Packet losses in internet protocol (IP) networks due to fading, congestion or network problems.

1.2.2 Different Approaches to Recovery/Restoration of Images with Lost Segments

There are three broad approaches for mitigating the loss of quality in received images due to packet loss:

- (a) Request for retransmission of the lost packets;
- (b) Error control via forward error correction (FEC) methods;
- (c) Error concealment (EC) methods.

These three methods are explored in some detail in Chapter 2. The focus of this thesis is the receiver-based EC image processing methods that aim to replace the lost packets with estimates obtained from the received ones. To recover lost packets from the neighbouring pixel values, EC methods utilise the structural observation that

images often contain high spatial correlations, recurring textures and patterns (Salama, Shroff & Delp, 1998; Wang *et al.*, 2002; Kim, Koo & Jeong, 2006; Zhai *et al.*, 2010; Asheri *et al.*, 2012).

1.2.3 Standard Test Images and Restoration Performance Measures

A standard test image is a digital image file used across different institutions to test image processing and image compression algorithms. Standard test images are maintained primarily to support research in image processing, image analysis, and machine vision. Therefore those images are used throughout this research to compare results, both visually and quantitatively with state of art methods. USC-SIPI Image Database is an example which can be found at <http://sipi.usc.edu/database>.

To evaluate the results, the restored image should be compared with the original. Two most widely used and popular objective image quality metrics are Mean Squared Error (MSE), and then Peak Signal to Noise Ratio (PSNR), which can be obtained from MSE (Thung & Raveendran, 2009). MSE is calculated as the average of the squared difference between the original image and the restored one. PSNR is commonly used to measure the quality of the restored image and illustrates the ratio between the maximum possible power of signal and the power of noise. The signal in this case is the original data and the noise is the error introduced by processing artifacts. The PSNR is selected to compare results of the proposed method with the previously proposed method as they used PSNR for evaluation.

The structural similarity (SSIM) index (Wang *et al.*, 2004) is an alternative method of quality measurement which is used in some parts of the thesis. This method is utilised for measuring the similarity between two images, with the measuring of image quality being based on a distortion-free image as reference. The mathematical expressions for performance measures are introduced in Chapter 3.

1.2.4 Image Processing Challenges in Image Gap Restoration

The main challenges in image gap restoration are listed as follows:

- *Selection of Transformation domain:* A decision needs to be made on the selection of the appropriate processing domain such as the raw image, frequency domain, wavelets, and pyramid. For practical reasons, as explained later, efficiency of performance, computational complexity and compatibility, pyramid discrete cosine transform (DCT) and wavelets were selected as the alternative pyramid transformation domain for image gap restoration.
- *Texture processing:* Estimation and reproduction of local texture information from the known samples in the neighbourhood of the gap. Textures are relatively homogenous patterns within the boundaries of objects or segments, they may involve local edges or slowly-varying changes in colours, but do not include major segment boundaries.
- *Structure/Pattern processing:* Estimation and reproduction of global, structural, information. Such structures include different textures and edge lines/curves that need to be modelled and reproduced in gap estimation.
- *Edges detection:* Estimation of local, global edges and segment boundaries. The estimation of ‘local’ textures or ‘global structures’ requires tools for edge detection and estimation.
- *Development of a hierarchical and/or iterative method:* For improved performance, in particular, when replacing large size gaps. The method combines pyramid transforms, a local texture model and edge detection for improved gap estimation.

1.2.5 List of Contributions to Image Restoration

1. Incorporation of edge-guided interpolation within multi-scale pyramid transformation for image gap restoration, which involves the integration of

edge-detection and segmentation of image objects. Subsequently, the interpolation is confined within homogenous regions to avoid blurring across edges and loss of edge information.

2. Inclusion of local edge or texture within pyramid image restoration. Global edge detection would be erroneous without first interpolating the gaps with a local texture interpolator as an initial approximation. Hence, local texture interpolation benefits are:
 - 2.1 Interpolation of textures within segmented homogenous regions;
 - 2.2 Pre-processing for subsequent edge-guided interpolation across ‘global’ segments;
 - 2.3 Can be used in strategies that combined local and global interpolations.
3. Inclusion of global edge estimates and interpolators within the pyramid transforms image restoration.
4. Comparison of the impact of the use of different edge detection and interpolation methods, as well as applying iterative method.
5. Investigation of a number of different wavelet types as an alternative to DCT pyramid transforms in edge-guided image gap restoration. In addition, the use of horizontal, vertical and cross details components of the wavelets for restoration of edges in images.

1.3 Organisation of the thesis

This thesis is organised into seven chapters and the six remaining ones are as follows.

Chapter 2: contains an overview of the published literature on the recovery and restoration of lost gaps in images. Image gap replacement has a wide range of applications, including: in-painting of missing or damaged segments in still/moving images (Padmavathi , Priyalakshmi & Soman, 2012; Swati, Malviya & Lade, 2013), replacement of image data packets lost in transmission, enhancement of distorted biomedical signals (Prochazka, Vysata, & Jerhotova 2010; Jonic & Sorzano, 2011), restoration of archived damaged images (Khoshelham & Elberink, 2012) and packet loss concealment over the internet protocol (IP) or wireless networks (Zeng & Liu, 1995; Banham & Katsaggelos, 1997; Rane, Sapiro & Bertalmio, 2003).

Different approaches and methods for correcting or minimising the degradation of transmitted image quality due to lost blocks of pixels are critically.

The advantages and disadvantages of various techniques are discussed. Moreover, insight regarding why some applications and tools were selected for developing the methodology for this thesis, is provided. The emphasis in the literature review is on published work in gap concealment which is the focus of this thesis.

Chapter 3: provides an overview of the theory and applications of image processing methods and tools used in this thesis, with the focus being on the following:

- Image processing basics;
- Discrete cosine transforms (DCT);
- Discrete wavelet transforms (DWT);
- Edge detection.

This chapter cover all mathematics which is used throughout the proposed method.

Chapter 4: an overview of the multi-scale pyramid image transformation method presented and applied to image gap restoration. Through a process of pyramid transformation and down-sampling, the image is transformed into a series of progressively reduced size layers until at the pyramid apex the gap size is one sample. The process is then reversed; at each stage, the missing samples are estimated, up-sampled and combined with the available samples.

This chapter includes an investigation into the different types of gap interpolation techniques and their advantages and disadvantages. The proposed algorithm includes a combination of multi-resolution transforms, different interpolation methods and blending techniques (DCT based interpolation) capable of restoring missing macro blocks. The algorithm is evaluated on various test images and the results are presented and reviewed in details.

Chapter 5: presents the second proposed methodology of image gap restoration, which is based on incorporation of edge-based directional interpolation within a DCT multi-scale pyramid transform. Two types of image edges are reconstructed: (a) the local edges or textures, based on the linear and locally-directional interpolation approaches, are inferred from the gradients of the neighbouring pixels and used to recover the missing gap, and (b) the global edges, or boundaries between image objects or segments, inferred using different types of edge detector applications. The evaluation is performed for the same images and the results are discussed in detail, being subsequently compared to those of chapter 4 and a range of published works.

Chapter 6: contains the third proposed methodology, a multi-scale pyramid method using wavelet transform, proposed as an alternative to the DCT-pyramid image gap reconstruction described in Chapter 5. The wavelet pyramid incorporates, as alternatives, conventional and edge-guided interpolation. The method is tested on the same set of test images and the results are discussed as well as being compared with those of DCT-pyramid interpolation and several state of the art methods, in relation to their capacities for restoration of the corrupted regions in damaged images.

Chapter 7: this final chapter concludes the work covered in this thesis and reviews the methods and results presented in the various chapters. The discussion in this chapter provides a summary and overview of several problems encountered in image gap restoration and consideration of how the proposed methodologies can overcome those problems. In addition, ideas for future work and extension of the methods proposed in this thesis are put forward.

Chapter 2

Literature Review

Chapter 2

2. Literature Review

2.1 Introduction

This chapter provides an overview of the published literature on the recovery or restoration of gaps in digitised images. That is replacement of pixels that are missing, lost or obliterated by noise. Image gap replacement has a wide range of applications including packet loss concealment over the internet protocol (IP) or wireless networks.

With the growing demand for image data streaming over the internet protocol (IP) networks, maintaining acceptable quality of service (QoS) is an important requirement for IP service providers, because these networks are ‘best effort environments, meaning that they do their best but cannot guarantee acceptable QoS. Three factors usually affect the QoS: jitter, delay and packet losses. Buffering and caching is used at the receiver side to conceal the effects of jitter and delay, however, in the case of packet losses the negative effect still remains (Hayasaka, Gamage & Miki, 2005). Packet loss can occur through network congestion, signal fading and thermal noise. Irrespective of the reason, data corruption degrades the quality of the received image and adversely impacts on the user experience (Kwok & Sun, 1993), as demonstrated in Figure 2.1.

This chapter starts with a review of three main approaches to minimising the degradation in quality of transmitted images due to lost blocks of pixels, namely: retransmission, forward error coding and error concealment (EC). It is argued that spatial error concealment (EC) methods have distinct advantages over the other two, in terms of requiring less bandwidth, having less delay, probable prevention of

additional errors and the ability to market as standalone or embedded applications. As a consequence, error concealment is the main subject of this thesis.

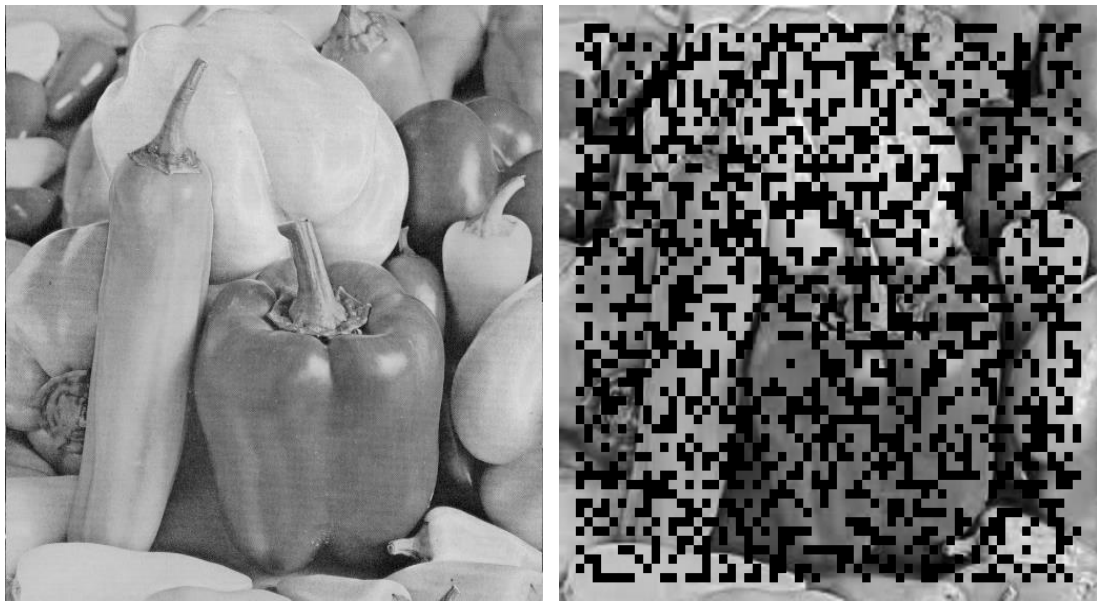


Figure 2.1: The original image of peppers and right, the image with 40% random missing macro blocks (8×8).

In particular, the focus is on the different strategies for implementation of EC techniques, such as the choice of front-end image transformation domains and the interpolation methods used to achieve improved error concealment. Hence in the relevant literature, in relation to the subject matter, discrete cosine transform (DCT) pyramid methods, discrete wavelets transform, edge-guided interpolators and edge detection methods are reviewed.

2.2 The Packet Loss Recovery and Concealment Methods

One of the most important applications of image gap restoration is packet loss concealment in transmission networks (Salama, Shroff & Delp, 1998). These packet loss errors might occur due to network congestion, thermal noise, switch noise or signal fading. The dominant cause varies according to the network and transmission media, such as landline and mobile devices. Additionally, as most signals transmitted on ‘real world’ channels are compressed, any packet loss can have a significant impact on the perceived quality of the decompressed images (Kwok & Sun, 1993).

There are three broad approaches for mitigating the loss of quality in received images due to packet loss, which are: (1) packet retransmission strategies (Feamster & Balakrishnan, 2002), (2) pre-processing error control coding techniques (Hayasaka, Gamage & Miki, 2005) and (3) post-processing error concealment algorithms (Zhai *et al.*, 2010). These are more formally known in algorithmic terms as:

- Automatic request for retransmission (ARQ) of the lost packets;
- Error control coding via forward error correction (FEC) methods;
- Error concealment (EC) methods.

2.2.1 Automatic Request for Retransmission (ARQ)

This method, on a request from the receiver, retransmits the original copy of the damaged/lost packet, as shown in Figure 2.2, which results in: recovery of lost images at the expense of an increase in the bandwidth and a growth in the delay proportional to loss rate (Hayasaka, Gamage & Miki, 2005). Clearly, this method can be used on request for retransmission in networks where is the provision of specific channels and protocols for an ongoing interaction between the sender and receiver modules.

In ARQ, the sender will retransmit the lost packets after receiving a notification of loss from the receiver, the so-called **negative acknowledgment (NACK)**, regarding the packet loss (Figure 2.2). However, this can involve a relatively high delay time for retransmitting the packets and hence, this method is unsuitable for real-time application. In addition, the act of retransmission can increase the congestion rate in an already congested network and as a result, more packet loss might occur. Hence, there needs to be a controller module for regulating the transmission rate.

To overcome the congestion problem, a source rate control method is used to give priority to the retransmit packets in order to ensure that they definitely arrive at the receiver end. When the network is congested, the controller reduces the source rate, thereby the encoding media transmission rate is reduced. For this purpose, a method of priority coding of the important part of the data has been proposed by Floyd, Padhye & Widmer (2000), where, the sender explicitly adjusts its sending rate as a

function of the measured rate of loss events, where a loss event consists of one or more packets dropped within a single round-trip time.

For dealing with the disadvantages of delay due to retransmission, a combination of receiver post-processing and selective retransmission has been proposed for the recovery of packet loss in image/video streaming over the internet (Feamster & Balakrishnan, 2002). The proposed method retransmits only the most important data to the receiver instead of all of the missing parts, such that from a set of the least retransmitted data a minimum level of quality of service can be constructed.

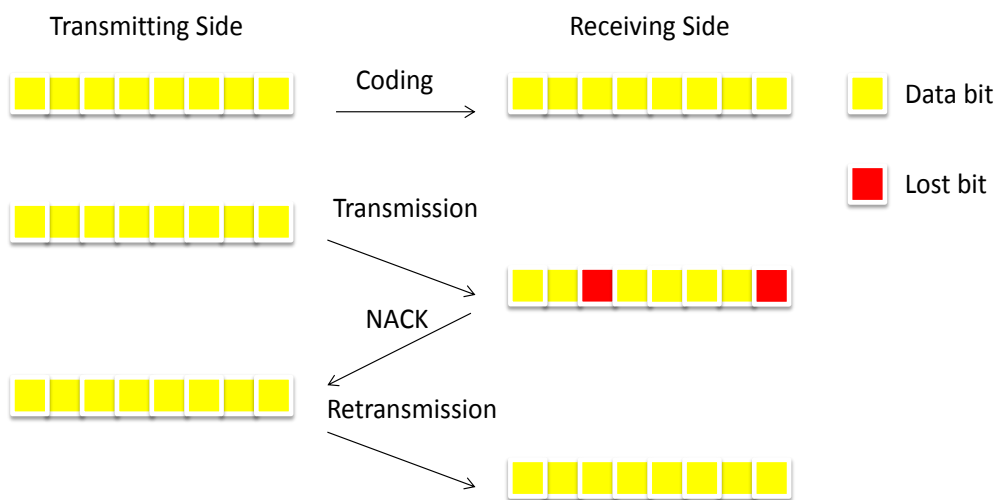


Figure 2.2: Framework of the automatic request repeat (ARQ) process, from the top row: with no error, with errors and retransmission.

A potential problem with the use of ARQ is due to the required enabling facility between the sender and receiver in that it is vital to have the retransmission protocol, two-way connection and additional bandwidth. However, in some error-prone networks, such as the user datagram protocol (UDP), there is no suitable communication channel for ARQ between the sender and receiver. As a consequence, the functions of acknowledgement of receiving data packet or request for retransmission are not available. The other drawback of this method can be seen in networks that are not immune to errors being repeated, such as the Internet, i.e. there may be recurrence of packet loss after retransmission in such networks. In addition, in Wang & Zhu, 1998 and Zhai *et al.*, 2010 the authors have shown that it might not be practical to retransmit the data for packet loss recovery in some time-sensitive applications, because each retransmission adds an additional round-trip

time to the system and for those applications that are time-sensitive any additional delays can disrupt all data and thus make them useless (Belfiore *et al.*, 2003). In summary, ARQ is not an efficient method for image loss replacement given all of the above mentioned drawbacks.

2.2.2 Forward Error Correction (FEC)

The second category of methods, FEC, employs error correction coding, with the addition of redundant ‘guard’ bits, to enable the recovery of lost pixels from the received information. Typically, in an (n, m) coding scheme, $k = n - m$ ‘guard’ bits are added to m data bits, usually using some form of linear block coding or convolutional coding algorithms. This implies that the pixel values in successive blocks of images will be coded, combined and/or spread over several successive packets. An example of (n, m) block coding is shown in Figure 2.3, where there are eleven packets in total: seven original data bits and four guard bits.

FEC codes increase the minimum distance between the raw source codes and hence, in proportion, to the increase in the minimum distance, increase the immunity to noise or data loss. Best known examples of FEC include Hamming (7,4) block codes with a minimum distance of 3 bits and error correction capability of 1 bit.

With an FEC method, all redundant data packets are transmitted with the original data packets (Hayasaka, Gamage & Miki, 2005). The advantage of this method is that it can guarantee correction of lost bits over less than the minimum distance, whilst the disadvantages are an increase in the number of data bits, greater bandwidth and delay (Ira, 2008; Yang & Bourbakis, 2009).

When FEC is employed in the whole network, it might increase the congestion owing to its overhead. Hence, this method is mainly useful in networks that have low congestion, because in highly congested networks with a high rate of packet loss, FEC can add extra traffic into the system, which will lead to a high amount of loss and consequently, the QoS will be degraded (Wang & Zhu, 1998).

To overcome the congestion problem, as a variation of FEC, an information hiding method has been introduced by Yang & Bourbakis (2009), whereby an additional

communication channel capacity has been added to the network from the data embedding capacity of the host. The embedded data is then extracted in the receiver to recover packet loss. There are three main parts to this technique: high bitrate information hiding, network modelling, and finally applying the error recovery technique.

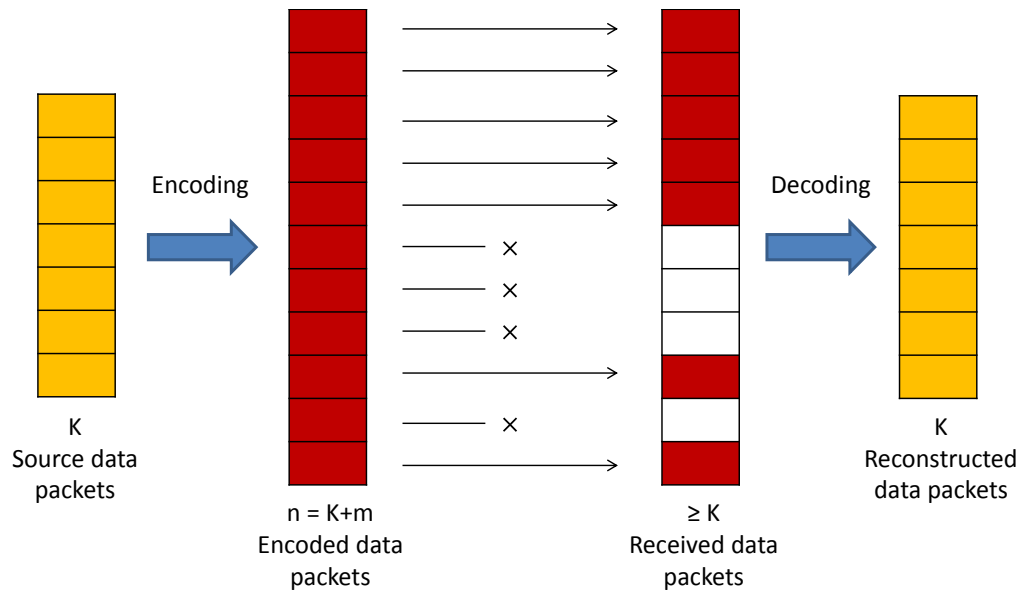


Figure 2.3: Diagram of the forward error correction (FEC) method (× show lost bits).

Hayasaka, Gamage & Miki (2005) proposed another variation of the FEC method to overcome some of its disadvantages. As buffer overflow at the routers side is one of the significant problems in an FEC, the aim of this method is to decrease the congestion caused by the overhead, by sending all FEC content to the receiver-end before the streaming starts. To make the method more efficient and cope with the overflow, priorities are also used to categorise the packets, with some being given low priority, based on the encoding technique and the FEC recovery features. The simulation results show improvement when compared with a simple FEC.

Ira, (2008) provides an algorithm called Lost Packet Recovery (LPR) to avoid packet loss by temporary allocation of a part of the bandwidth for Forward Error Correction. Once packet loss happens, a part of the bandwidth is allocated for sending the FEC data to the receiver end and the size of the FEC data channel is changed iteratively to find the minimum bandwidth required for the receive system to reconstruct all lost parts. As the available bandwidth in network is usually limited, another version of

FEC, called priority encoding transmission, was proposed by Albanese *et al.* (1996), to prevent a high level of bandwidth requirement. This is based on giving priority value to each part of the bit-stream. The priority values specify the sufficient amount of correctly received packets so as to recover the missing parts at the receiver side. Consequently, the receiver is still capable of restoring the missing parts when it has received a sufficient amount of packets. Whilst this method decreases the data rate, it is not capable of dealing with burst losses.

FEC is an end-to-end (from sender to receiver) method for packet loss recovery, but when it is applied to all available bandwidth then the additional overhead is added to the network and the QoS may decrease. Most of the losses are due to buffer overflow at the routers which is caused by network congestion. So, a method that does not add extra overhead to the network is preferable.

2.2.3 Error Concealment (EC) Method

The third category of methods, EC, involves receiver-based image processing aimed at replacing the lost packets with computational estimates obtained from the correctly received data (Banham & Katsaggelos, 1997). To recover lost packets from the neighbouring pixel values, EC methods utilise the observation that images often contain high spatial structures, correlations between neighbouring pixels as well as recurring textures and patterns (Suh & Ho, 1997; Wang, Yu & Zhang, 1998; Hemami, 1995; Rane, Sapiro & Bertalmio, 2003; Agrafiotis, Bull & Canagarajah, 2006; Zhai et al., 2010).

In general, these methods are divided into temporal and spatial approaches depending on the type of information that they use. The former is suitable for moving images and videos, employing the temporal information from consecutive frames, whilst the latter utilises the spatial information from intra coded frames or images. For videos, there are also methods that combine the temporal and spatial approaches.

Further development of spatial EC for the recovery of missing image blocks is the chosen focus of this research thesis, due to following assessment:

- Among the three aforementioned categories of solutions for packet loss recovery, an effective EC would be most beneficial as it adds no further load onto bandwidth or causes delay;
- EC methods can be used for real-time or non-real-time applications;
- In contrast to EC, ARQ is not compatible for real-time applications, as it causes irregular delays in packet delivery;
- In contrast to FEC, EC does not require adding redundant information to the data, which leads to use of more bandwidth, an increase in delay and a decrease in the compression ratio;
- The ARQ and FEC techniques are not immune to further errors;
- Unlike EC, ARQ and FEC are not standalone applications; EC can be embedded or downloaded as a stand-alone app for use at the receiver terminal;
- In contrast to ARQ and FEC, EC methods do not require an international telecommunication union (ITU) approved standard and as mentioned, can be coded as stand-alone apps and deployed in networks or used as embedded applications on the receiver handsets/terminals.

Therefore, for above reasons advantages which will be used later in these research, spatial EC image gap restoration is the category of solution explored in this research.

At their core, image EC computation algorithms often involve three distinct processes:

- Front-end image transformation;
- Structural data extraction, such as segment boundaries or edges;
- Extrapolation or interpolation of missing gaps.

The varieties of image transformations used are many, with the most popular being the identity transform (raw image), discrete Fourier transform (DFT or its fast version FFT), discrete cosine transform (DCT) and the discrete wavelets transform (DWT).

The main structural information used for EC is segment boundaries obtained by use of edge detection, which are employed as part of edge-guided interpolators. There are also various types of spatial EC restoration methods, such as linear or averaging

interpolation (Shirani, Kossentini & Ward, 2000), bilinear interpolation (Salama, Shroff & Delp, 1998), directional interpolation (Asheri *et al.*, 2012) and statistical learning based interpolation (Zhai *et al.*, 2010), with the first two methods being defined as non-directional, whilst the rest are based on directional interpolation categories.

2.3 Image Transformation Methods for Gap Interpolation

At the front-end of a spatial EC method there is usually an image transformation module. The selection of the transform domain in which the image is interpolated can have a major impact on the methodology and outcome of the restoration technique (Park, Kim & Lee, 1997; Alkachouh, & Bellanger, 2000; Shirani, Kossentini, & Ward, 2000; Meisinger & Kaup, 2004; Park *et al.*, 2005). The common choices of transform domain vary from direct interpolations over raw spatial domain pixels (Kaup, Meisinger & Aach, 2005; Asheri *et al.*, 2012), to methods that involve transformation of images using discrete Fourier transforms (DFT) (Meisinger & Kaup, 2004), discrete cosine transforms (DCT) (Wang & Zhu, 1991; Zhai *et al.*, 2010) discrete wavelet transforms (DWT) (Rombaut, Pizurica, & Philips, 2008; Prochazka, Vysata & Jerhotova, 2010) or EC methods using a combination of spatial and frequency domains as investigated in Wang, Zhu & Shaw, 1993, Alkachouh, & Bellanger, 2000, Zhai *et al.*, 2010 and Marvasti *et al.*, 2012).

2.3.1 Discrete Cosine Transform (DCT)

DCT transforms are the most widely used frequency transforms in image coding applications (Wang & Zhu, 1991; Zhai *et al.*, 2010). There are three reasons for the common use of these transforms in image processing:

- (1) *Optimality*: The fixed DCT matrix efficiently approximates the performance of data-dependent optimal Karhunen-loeve transform (KLT) (or equivalently principal component analysis (PCA)) derived from the covariance matrix;
- (2) *Practicality*: DCT is relatively easy to implement, it is robust and computationally efficient;

(3) *Compatibility*: DCT is already the standard transform employed for many image coding apps such as Jpeg and Mpeg.

Hence, unsurprisingly, DCT are used in most research and development efforts (Wang & Zhu, 1991; Zhu, Wang & Shaw, 1993; Wang, Zhu & Shaw, 1993; Alkachouh, & Bellanger, 2000; Zhai *et al.*, 2010) as a solution to recover the corrupted areas of an image. These DCT-based methods advantageously utilise the observation that images, except for the edges, have primarily low frequency content. Hence, the high frequency coefficients do not carry important image information and most image information is compressed into low-frequency coefficients (Park & Lee, 1999).

Frequency domain reconstruction is applied to gap restoration in Wang & Zhu (1991). A linear combination of correctly received coefficients is calculated within the surrounding blocks to replace the missing areas. However, this method only reconstructs the lowest fifteen coefficients for a complete block loss, in the case of an 8×8 macro block size (64 coefficients) and the rest of the coefficients remain lost (zeroed). Hence, most of the high frequency details are lost and so the quality of the image is degraded.

Alkachouh & Bellanger (2000) proposed an approach for replacing a missing block by using a DCT, which transforms the missing block and the available neighbouring pixels into the frequency domain. The higher frequencies are set to zero as the lower ones are deemed more important and carry the main information. Then, the same position as the missing block in the neighbouring blocks is used to recover the missing data areas. Perhaps unsurprisingly, the PSNR outcome is not convincing, especially regarding the details and diagonal edges, owing to the high frequency information having been removed.

In Wang, Zhu & Shaw (1993) the maximally smooth image recovery method was put forward using a constrained energy minimisation approach in the DCT domain. It calculates the spatial variation (gradient or Laplacian) between pixels in the missing block and the adjacent pixels in the neighbouring blocks, subsequently, minimising this measure to recover the lost blocks. However, as this scheme is based on a low-pass filter, it ignores high frequency data and consequently, the edges, the

boundary areas and lines are adversely affected. Also, due to utilising only a small part of the available image area, this technique is not able to provide accurate structural/pattern information in the image reconstruction.

A combination of frequency domain and spatial interpolation was used in Hemami & Meng (1995), based on the assumption that there is a similarity between lost coefficients and the corresponding ones in the neighbouring blocks. Four corresponding coefficients in the surrounding blocks are selected to interpolate the value of a missing block by using the method of maximally smooth recovery in Wang, Zhu & Shaw (1993). Due to the distance of the selected coefficients for interpolation, from the missing pixels, the result might be inaccurate owing to the low level of correlation.

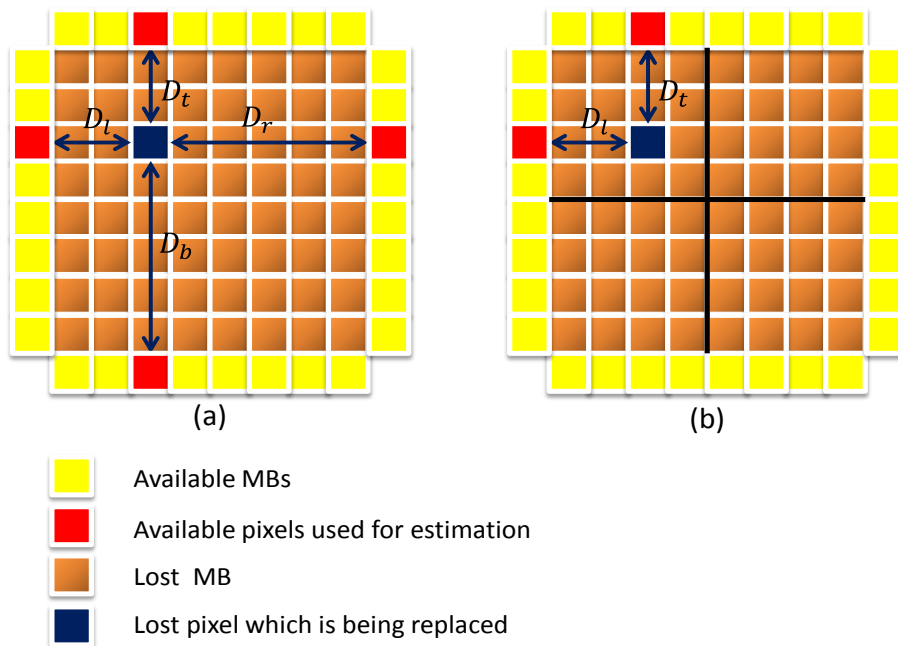


Figure 2.4: a) The missing pixel is interpolated by the values of four pixels on all the boundaries. b) The missing pixel is estimated by two pixels on the nearest two boundaries.

Aign & Fazel (1995) proposed a method of interpolating the missing pixels by utilising four immediate pixels in four neighbouring blocks, instead of the corresponding pixels, in contrast to the method used in Hemami & Meng (1995), which resulted in the estimation being more accurate. There are two alternative estimation models of the missing block in this method (Figure 2.4) with first, the

missing pixel being interpolated by values of four pixels on all the boundaries (Figure 2.4.a), whilst for the second model the missing pixel is estimated by two pixels on the nearest two boundaries (see Figure 2.4.b).

A further approach to restoration of a missing block using a DCT transform domain was proposed by Ancis & Giusto (1999), which investigated four different methods of interpolations. Among four considered techniques a combination of median and edge based interpolation was selected in order to recover the missing blocks.

2.3.2 Discrete Wavelet Transform (DWT)

As an alternative approach, discrete wavelet transform (DWT) can be used, instead of the discrete cosine transform (DCT), to reconstruct the corrupted image pixels in error prone networks. The essential differences between DCT and DWT are:

- (1) DWT employs a set of varying duration (scale) and frequency basis functions, with ‘bell-shaped’ decaying profile, known as wavelets, whereas the cosine basis functions in DCT have the same duration and are of constant amplitudes and have harmonically-related frequencies;
- (2) DWT progressively and equally divides bandwidth and transforms images into four frequency quadrants (Low-low, low-high, high-low, and high-high). Hence DWT inherently has a pyramid-like structure, which is useful as pyramid type transforms are employed in this thesis;
- (3) At each decomposition stage, DWT involves a set of different scales that may be better suited to non-stationary edge-type structures;
- (4) DWT is computationally more complex to implement and run than DCT.

Under this approach, the main assumption is that the correlation between the lost block and its surrounding ones can be evaluated in the wavelet domain to replace the missing parts.

The easiest way to cope with data loss in a DWT transform is to discard any that occurs and replace its coefficients with zeros. Consequently, the received image has dark ‘spot’ owing to the discarding of the low-frequency coefficients that contain most of the energy. As removing the lost coefficients causes image quality

degradation, to overcome the problem by a basic DWT gap restoration, a simple method of using four surrounding neighbouring blocks to calculate a linear interpolation has been proposed by Bajić & Woods (2003). Even though the method is not complicated and the outcome is acceptable in the smooth area, inferior restoration occurs at the edges, as can be seen in Figure 2.5.



Figure 2.5: Recovered Lena image by Bajić & Woods (2003) with 25% packet loss.

Two separate methods were utilised to restore the missing coefficients in Hemami & Gray (1997), one for the low-frequency coefficients (smooth areas) and the other for the high-frequency coefficients (edges areas). The correlation of the available low-low sub-band at the lowest level of decomposition is used to recover the smooth areas (utilises both inter-band and intra-band correlation between the coefficients). In addition, a bi-cubic interpolation approach is applied to reconstruct the corrupted coefficients at high frequencies. High-low and low-high sub-bands are used to estimate the vertical and horizontal edges for each missing coefficient through thresholding at the same decomposition level and the lost high frequency coefficients in the HH sub-band being ignored and set to zero. This method has been tested only on uncompressed images and most of images in packet networks are compressed, hence it is not appropriate for widely used compressed image types.

Another method of DWT restoration was proposed by Rombaut, Pizurica, & Philips (2008), which involves an adaptive interpolation approach that takes into account the available coefficients in both the horizontal and vertical directions. In addition, an

optimal interpolation weight is calculated from the correctly received surrounding coefficients. Each lost low-frequency coefficient is restored by these interpolation weights, which are computed from interpolation errors. These errors are based on interpolation of the missing coefficients from available neighbouring ones in both vertical and horizontal directions.

In Lee & Chen (2002) an approach to DWT restoration was presented that utilises the correctly received bit-plane to recover damaged bit-plane data in the missing parts. However, as wavelet transform eliminates most of the correlation between coefficients, there is not sufficient information for restoration and hence, the published result of the proposed method is not convincing (as the outcome can be seen in that paper).

Ye, Sun & Chang (2004) proposed an edge-based filter to restore the missing coefficients in the wavelet domain. The reconstruction is based on a combination of the correlations in the spatial and wavelet domains. After estimation of the missing parts, a refinement algorithm is applied into the recovered coefficients, which involves two parameters; the first being the statistical correlation between the coefficients and the second, keeps the correctly received coefficients unchanged. More edges can be preserved by this method.

2.3.3 Pyramid Transforms

Pyramidal data coding/transform structure is an important transform coding scheme in the field of image processing with applications regarding image compression, restoration and recognition (Adelson *et al.*, 1984). This method is a transform or filter-based representation of a signal, which decomposes the input data into subsets of progressively reduced-scale, bandwidth, basis functions and coefficients information, at several levels of the pyramid. This is in order to extract the main features as well as reduce the noise and redundancy. The latter feature can be beneficial in the coding, image enhancement and image analysis. Furthermore, the human visual system includes a similar decomposition process (Tan & Ghanbari, 1992).

Pyramid transforms are structurally similar to wavelet transforms and the latter could be regarded as a category of the former. Gabor developed a form of pyramid (wavelet) transform by using a combination of complex sinusoids and Gaussian windows functions, which perform the optimal time-frequency localisation outcome (Lee, 1996).

One of the original papers on the application of the pyramid method to image processing pertains to the classical Laplacian pyramid coding, first proposed by Burt & Adelson (1983). Their method is based on the assumption that different scales of an image can be used as the basis functions, when those scales have the same basic shape but appear at many scales. Redundancies between pixels are removed by separating the low-pass version of the image from the original one and the low-pass filtered part appears in the next scale level. Then, the same process is repeated progressively on the low-pass filtered section, to reach the required level of decomposition.

As an alternative to the Laplacian transform, DCT transform has been employed in pyramidal coding (Tan & Ghanbari, 1992) as it is a widely used, efficient and robust transform and additionally has the important property of compacting most of the energy in the M lower frequencies coefficients from a total of N coefficients. Consequently, the $(N - M)$ DCT coefficients, which do not carry important information, can be omitted.

Figure 2.6 illustrates the decomposition of an image into a DCT-pyramid together with the reconstruction process. The basic operations employed are DCT, IDCT and subsampling/decimation, the latter amounts to retaining M_i out of N_i DCT coefficients. While the main theoretical restriction is $M_i < N_i$, for gap restoration a value of $M_i = N_i/2$, is used.

The processing functions that transform one pyramid level onto the next reduced-scale level are as follows:

- 1- Apply DCT to an N_i -pixels image at the current level to obtain N_i DCT coefficients;
- 2- Select M_i lowest index coefficients out of N_i coefficients;

- 3- Apply IDCT to M_i lowest coefficients to obtain the decimated image for the next level.

Decimation is used in pyramidal data coding to reduce the image and transform its size to the next level by retaining a subset of M out of N coefficients. This is done by applying an N -point DCT transform, the selection of a subset of M coefficients and subsequent application of an M -point inverse DCT transform. The pixels are derived by sub-sampling the filtered original image by an interval of $\frac{N}{M}$.

Once a pyramid of DCT coefficients, such as that shown on the left hand side of Figure 2.6, is constructed, the desired functions, such as quantisation, noise reduction or image restoration can be applied to each level of the pyramid. The original image can then be reconstructed from the decimated and processed sub-images by interpolation in the DCT domain. The process of interpolation (up-sampling) starts from the apex by applying an M point DCT transform on the decimated image and then zero padding from a size of M to the size of the next level N , followed by an N -point inverse DCT transform, as can be seen in Figure 2.6.

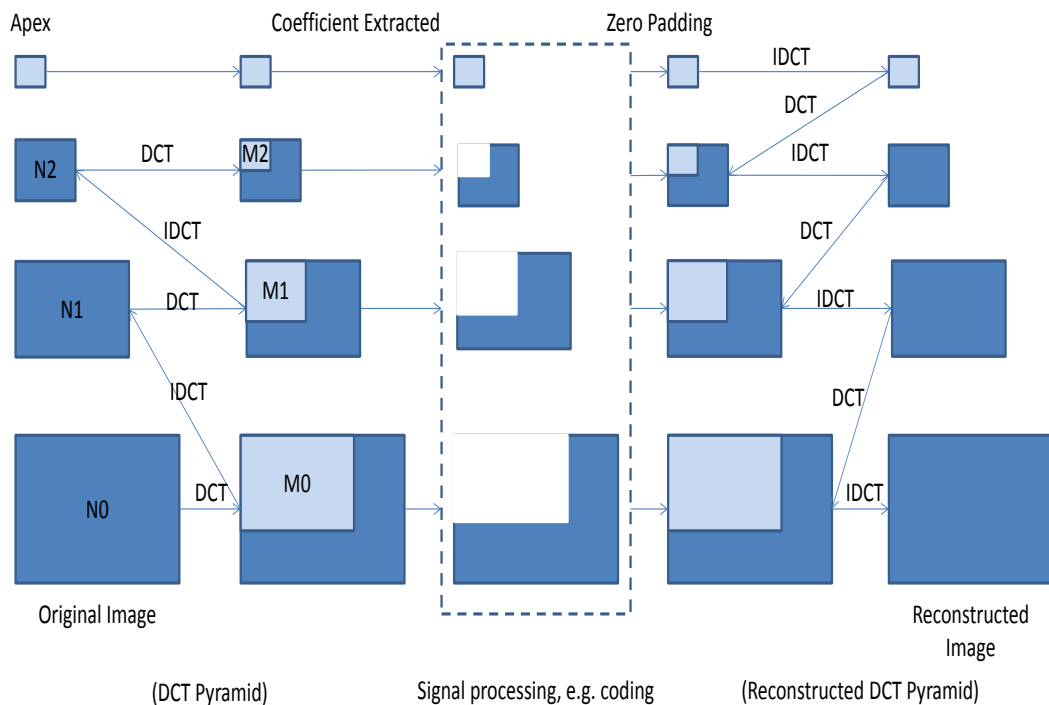


Figure 2.6: Illustration of DCT-Pyramid processing. Three levels of a two-dimensional DCT pyramid coding and reconstruction IDCT pyramid (Tan & Ghanbari, 1992).

Using the DCT transform helps to reduce the blocking effects and in addition, the drawbacks of Laplacian pyramid are resolved. First, in contrast to the Laplacian pyramid, the difference image is in the transform domain and second, the whole image part of the DCT pyramid always remains equal to A , albeit on reduced scale (the total image area).

2.4 Interpolation of Missing Gaps

Interpolation methods are used to replace unknown pixel values from the known data values. For interpolation, the choice is broadly between non-directional (Salama, Shroff & Delp, 1998; Wang *et al.*, 2002; Wang & Zhu, 1991; Wang, Zhu & Shaw, 1993; Zhu, Wang & Shaw, 1993; Park, Kim & Lee, 1997; Wang, Yu & Zhang, 1998; Alkachouh & Bellanger, 2000; Agrafiotis, Bull & Canagarajah, 2006) and directional (Kwok & Sun, 1993; Zeng & Liu, 1995; Hsia, 2004; Kim, Koo & Jeong, 2006; Asheri *et al.*, 2012) interpolation. Each approach has its own advantages and drawbacks.

Bilinear and non-directional techniques are able to recover the smooth areas, but fail to restore the visually important edge information. In contrast, directional interpolations are able to recover edges more accurately, but leave the blurring artefacts on the smooth areas.

2.4.1 Non-directional Interpolation Methods

One type of a non-directional method is basic spatial interpolation, which uses a weighted average of the neighbouring pixels to recover the lost gap. One of the weight-averaging approaches devised by Salama, Shroff & Delp (1998) employs four immediate neighbouring pixels in four directions. The missing pixel is replaced by the weighting composition of these four available pixels, but despite this method being simple, the result is not persuasive as the edges and structures cannot be accurately reproduced.

Varsa, Hannuksela & Wang (2001) utilised the same approach of the weighting of the average interpolation of four pixel values in four directions, top, bottom, left and

right, as can be seen in Figure 2.7, where $W1, W2, W3$ and $W4$ represent the distance between the interpolated pixel and the available neighbouring pixels in the top, bottom, left and right directions. It is assumed that all four corresponding pixels are available, but in real scenarios those pixels can be affected (Alejandro *et al.*, 2012) and if this is the case, then the quality of the restored image decreases significantly. A further shortcoming is that only the immediate neighbouring pixels are used for the interpolation and hence, the method does not include a wider area. This means it might not obtain all the information required for a high quality restoration. Although with this method and other weighting averaged approaches, a satisfactory result is achieved in the smooth areas, the performance around the edges can be blurred, as it does not model the impact of edge discontinuity (Hemami, 1995; Wang *et al.*, 2002).



Figure 2.7: Weighting average interpolation, one missing MB (8×8), four available surrounding blocks.

The shortcomings of the methods explained above demonstrate that it is essential to use a large ‘*global*’ area of surrounding MBs to reconstruct edges rather than just using the immediate local neighbours.

Zhu, Wang & Shaw (1993) proposed another method which is a combination of coding and EC for DCT-based packet images. The method relies on the hypothesis that image contents change smoothly as well as having a low frequency component and block interleaving was the solution they used. This involves gathering all spatially adjacent image blocks into distinct packets and maximising the distance between adjacent blocks. As placing the nearby blocks at a far distance leads to reconstruction/encoding delays, to overcome these, an even/add interleaving

program is sometimes used. Consequently, when a block with an even index is lost, its neighbour with an odd index is generally available. Afterwards, a smooth reconstruction process is applied to the lost areas to reconstruct the corrupted block by finding the best maximally smooth block that fits along the boundaries (Debrunner *et al.*, 2000). Whilst this method allows for the reconstruction of the smooth area, unfortunately, the resulting image is blurry at the edges. Furthermore, as these methods take account of both internal smoothness similarities in the block and boundaries, the computation complexity is high.

Hemami (1995) proposed two different approaches to restoring missing image blocks, the first involves the use of vector-quantised linear interpolation, while the second, employs a linear combination of four neighbours of the missing macro block's weights. However, a 10% space overhead is added to the system by the former method and the reconstruction of the main edges are not convincing when applying the latter.

To overcome the disadvantages of (Wang & Zhu, 1991; Zhu, Wang & Shaw, 1993; Wang, Zhu & Shaw, 1993) algorithms, Park, Kim & Lee (1997) suggested a new method, which uses a smoothness constraint along the boundaries that involves solving a set of linear equations. The method reduces the computational complexity by decomposing the linear equation into four linear equations, thereby making the method more suitable for real time applications. To restore the corrupted blocks the cost function needs to be computed in the DCT domain and its derivation is calculated for each of the missing 64 coefficients of a macro block. It has been shown that this method is able to recover a maximum of 28 coefficients, which are mostly low frequencies and the rest are set to zero, consequently the high frequencies and edges are not recovered. Despite the new method utilising less restoration information, the quality of the restored image is better than those of the previous methods.

An approach by Shirani, Kossentini, & Ward (2000) extracts the correlation between adjacent blocks. This method utilises a weighed linear combination of all neighbouring available blocks (a least squares solution minimising the boundary error between the known and missing blocks) and the collection of this surrounding information leads to achieve a better estimation for diagonal edges.

Rane, Sapiro & Bertalmio (2003) presented a technique which utilises the correlation between the lost packet and its neighbours and it employs a combination of two separate algorithms. Firstly, all lost blocks are divided into texture and structure types. Then, texture synthesis is used to reconstruct the texture type missing blocks and an image in-painting method is deployed to restore the structure type missing blocks. However, there are several drawbacks to this method, for in the cases of losing the image feature completely or there being a high level of missing blocks, it is not able to recover the image properly. In addition, it is not always easy to predict the type of missing block, particularly when there is a high level of loss and therefore, methods with the capability of dealing with such loss are preferable (Meisinger, 2007).

A further approach to SEC is described in Wang, Yu & Zhang (1998) using a technique called best neighbourhood matching, which is based on block-wise similarity within the image (special kind of information redundancy). The method searches for the best similar macro-block (MB) within a predefined search area in the image in order to replace the missing one with it. The process, as shown in Figure 2.8, starts by defining the search area, called the centre point and then an extension is added to the missing block in the local neighbourhood in each dimension, which is called the range block. Afterwards, the search for a 'good' block in the image is started by using the appropriate luminance transformation in each block candidate. Then, the mean squared error (MSE) distance between the shifted and centre blocks is calculated for each of the candidates of the same size and the block with the smallest MSE, called domain block, is selected to fill the missing part.

Although it has been proven that the reconstructed image quality is acceptable in Wang, Yu & Zhang (1998), the running search time is high, especially regarding use in real-time applications. Some work has been done to improve the algorithm, for example, by He & Zhang (2010) aimed at reducing the runtime and computational complexity by deploying a rotating style, which dynamically detects and cuts the search range.

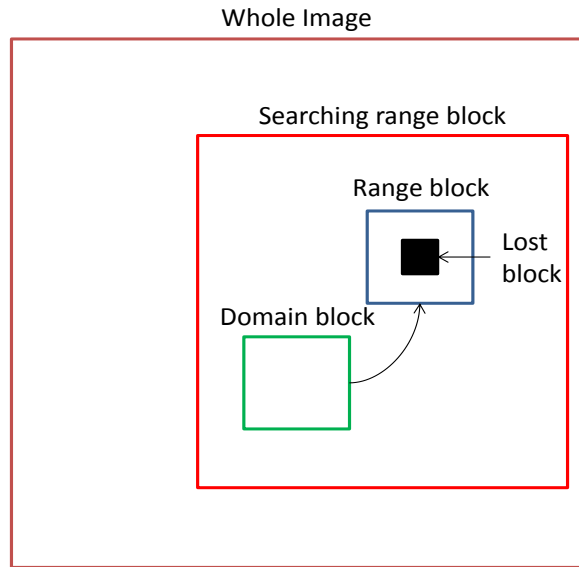


Figure 2.8: Diagram of a lost block, range block, domain block and searching range block.

Another work of block-level similarity put forward by Zhai *et al.* (2008). They established a block-based bilateral filtering (BBF) framework, which operates in a block-wise manner. It has been shown the problem of error-concealment using BBF can be considered as a superset of image denoising using BF. The BBF has the ability to capture the block-level similarity that well matches the need of error-concealment.

Finally, in Tschumperle & Deriche (2005) vector-valued image regularisation based on variation methods and partial differential equations were introduced for image enhancement and in-painting. This formulation is particularly adapted to understand the local smoothing behaviour of diffusion partial differential equations (PDEs).

2.4.2 Directional Interpolation Methods

From the research literature on image processing, it is evident that EC methods work well in recovery of the low frequency smooth parts of the image (Wang & Zhu, 1991; Wang, Zhu & Shaw, 1993; Zhu, Wang & Shaw, 1993), but they are normally incapable of restoring the high frequency details and also, they miss the main edges. Consequently, to obtain high quality image restoration results, it is of paramount importance to include the image edge information explicitly.

In directional interpolation, a more complete utilisation of the spatially correlated details from the neighbouring pixels is used to perform a better interpolation of the smooth and edge details present in the surrounding blocks. Many techniques have been proposed for using edge-related information for image interpolation (Kwok & Sun, 1993; Hsia, 2004; Zhao *et al.*, 2005; Kim, Koo & Jeong, 2006; Asheri *et al.*, 2012), with some employing simple approaches to estimate edges within images (Suh & Ho, 1997; Salama, Shroff & Delp, 1998; Park & Lee, 1999), which are capable of interpolating the low frequency details of missing areas, but they fail to reconstruct the highly detailed parts. Other approaches (Wang, Yu & Zhang, 1998; Li & Orchard, 2002; Park *et al.*, 2005; Gharavi & Gao, 2008) have achieved an improvement in image detail restoration and have provided more reliable interpolation results.

A directional interpolation approach was proposed by Kwok & Sun in 1993, for which a Sobel filter is used to calculate the gradient vector G for each pixel by employing the local geometric information from the correctly received neighbouring blocks to the missing one, which leads to the strongest edges being found. Then, the gradient angle is computed by the following equation to find the strongest edge direction:

$$G = \sqrt{g_x^2 + g_y^2} \quad \theta = \tan^{-1} \left(\frac{g_x}{g_y} \right) \quad (2.1)$$

Eight possible directions are defined by rounding the gradient angle to the nearest value of 22.5° equally spaced around 180° (see Figure 2.10). A counter is allocated to each direction and is incremented if a line of the magnitude gradient and angle θ at pixel $p = (i, j)$ is passing through the missing area.

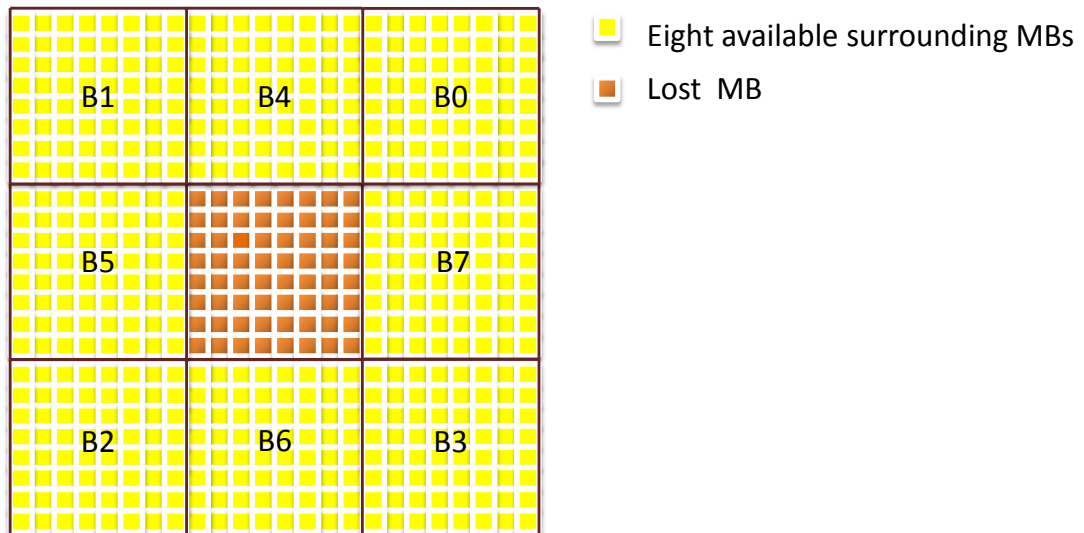


Figure 2.9: Missing MB (8×8) with eight available surrounding MBs (8×8).

In Kwok & Sun (1993) it is assumed that all eight neighbouring MBs are available (Figure 2.9) to calculate the missing MB and hence problems may arise if any are absent. Edge detection is a non-trivial process and any error can yield noticeable artefacts in the image restoration outcome. Moreover, as the method only takes into account the immediate surrounding MBs, the performance suffers if the predominant edge map is different to the immediate surrounding MB area (Alejandro *et al.*, 2012).

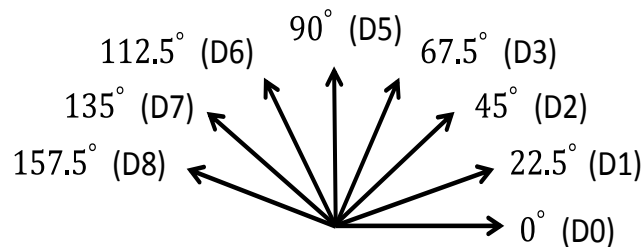


Figure 2.10: Eight edge directions.

Two factors are involved in an alternative method proposed by Jung, Chang & Lee (1994) for interpolation: interpolation direction and projection data. Missing blocks are estimated from correctly received neighbours through bilinear reconstruction, which is followed by an enhancement step. Projective interpolation uses the edge pattern enhancement to improve the edge restoration. The quality of outcome is improved (as can be seen in (Jung, Chang & Lee (1994))) and this is mainly due to the good estimation of the edges.

Projection onto convex sets (POCS) is another method employed by Sun & Kwok (1995) (see Figure 2.11), which utilises band-limited extrapolation by spectral estimation in addition to the gradient measures in the spatial domain, as with the previous method. Neighbouring blocks are divided into two categories: smooth and edge areas. A discrete Fourier transform (DFT) module transforms the missing blocks and their available neighbouring ones together as one large block. Then, separate transforms are applied to each block based on its category. A low-pass filter and a band-pass are used for the smooth and edge areas, respectively. Afterward, the neighbouring blocks are replaced with estimated samples after inverse DFT. When multi-directional edges exist in the missing parts, the missing block is interpolated along all the edge directions and then the method utilises the prospective principle to unify these multiple edges, as can be seen in Figure 2.11.

As explained above, the process includes subsequent iterations and also two transforms are needed for each, consequently the computational complexity is high. Moreover, this method is not able to restore multi-directional edges, the smooth areas are not immune to incorrect restoration owing to overshoot effects caused by band-limited extrapolation methods (Meisinger, 2007) and the edge detection operators are sensitive and might give false detection in the edge areas, which leads to wrong block classification. Generally, this method is usually reliable for situations where all the surrounding MBs are available, but in cases where the neighbouring blocks are corrupted, the resulting image is not satisfactory.

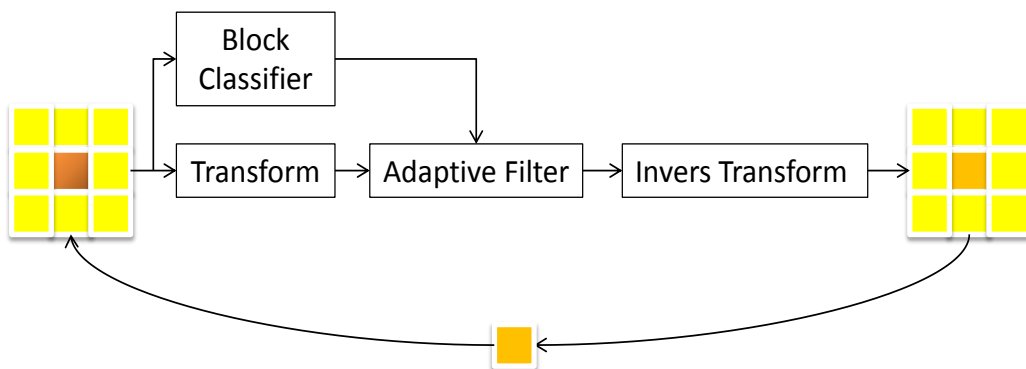


Figure 2.11: POCS iterative process on a missing block and eight available surrounding MBs (Sun & Kwok, 1995).

In Zeng & Liu (1995), a spatial directional interpolation scheme is proposed, which makes use of the local geometric information extracted from the surrounding blocks.

A neighbouring frame, two-pixels wide, from available surrounding pixels is used in the proposed method to restore the missing block. A geometrical structure is exploited from that frame by first thresholding to a binary format, then by using the number of changes from black to white and also the location on the frame, the edges and their directions that are passing the missing block, are estimated. Afterward, based on edge and direction estimation the block is divided into different zones and different types of interpolation are employed to interpolate the missing block.

To overcome the drawbacks of the previous methods, Rabiee, Radha & Kashyap (1996) proposed a scheme called multi-directional recursive nonlinear filtering (MRNF) (see Figure 2.12). The interpolation starts from the boundaries of the MB moving towards the centre and the correlation between it and its neighbouring MBs is extracted by a variable kernel, which works as a processing window. While in another exploration the relationship between surrounding macro blocks is exploited by using a W-GMLOS (weighted generalised maximum likelihood ordered statistics) filter. This combination leads to a probing of the cross-correlation between the corrupted macro block and its neighbouring ones.

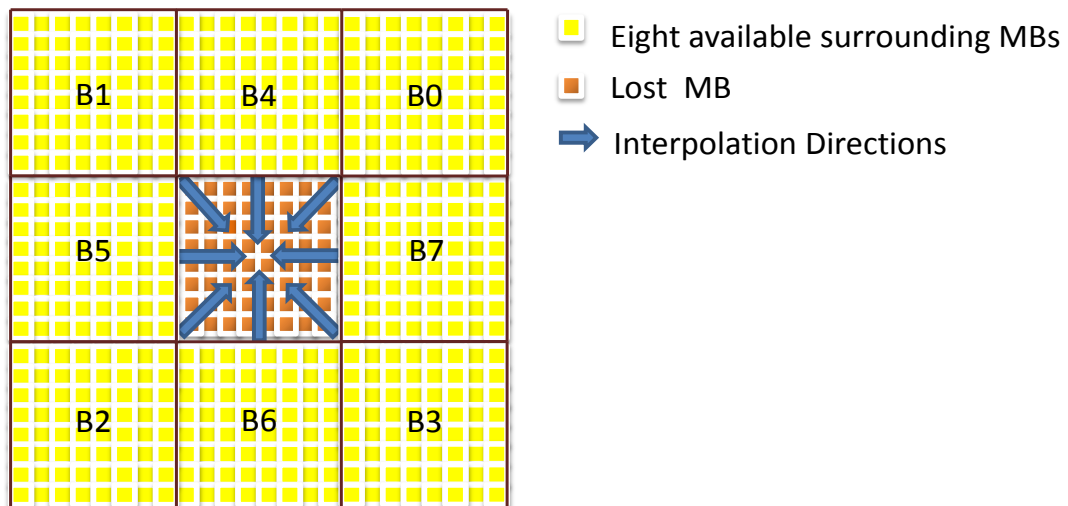


Figure 2.12: Multi-directional recursive nonlinear filtering (MRNF) and eight interpolation directions.

Park & Lee, (1999) presented a two-dimensional non-uniform rational B-spline (NURBS) interpolation process. After employing this function, this method involves utilising an optimisation technique to find the optimal control point so as to be able to replace the missing area. The sharp edges and smooth areas can be estimated more intensely, as this technique uses all the information along surrounding neighbours.

Despite the outcomes being improved compared with the previous proposed methods, which just use bilinear (Salama, Shroff & Delp, 1998), DCT transform (Wang & Zhu, 1991) Projection onto convex sets (POCS) (Sun & Kwok, 1995), the quality is still unconvincing, as can be seen in Figure 2.13. In addition, it requires hundreds of iterations to find the optimum result and hence, involves a high amount of computation complexity.

Hough transforms and edge detection were combined by Gharavi & Gao (2008) to propose a method for edge detection. The scheme is based on designing a new Hough transform-based technique that is capable of continuously connecting edges irrespective of the number of edge points surrounding missing areas. Then, the connected edges are used to divide the missing areas into different regions for interpolation along the directions of each detected line. However, the performance result of the method decreases in the case of an image with multiple edges and texture.

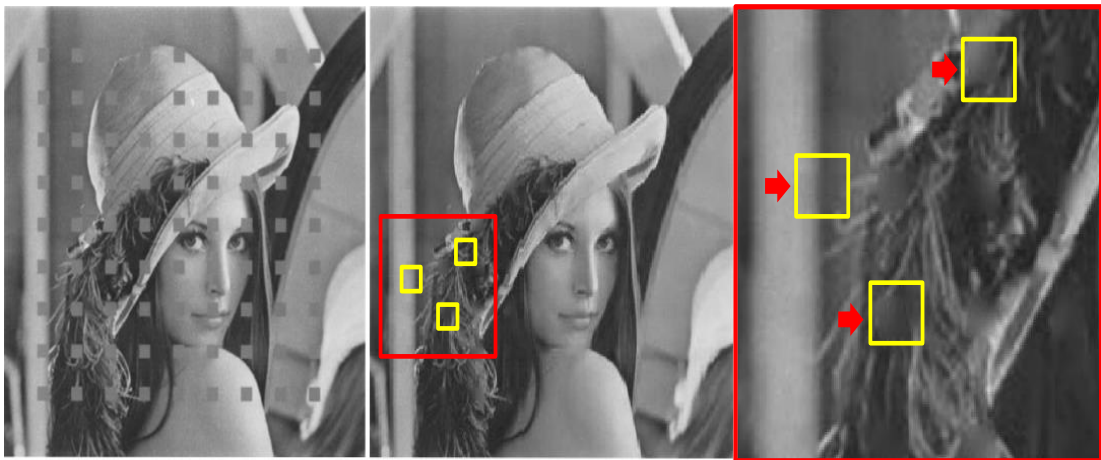


Figure 2.13: Corrupted Lena, the restored Lena by (NURBS) interpolation and the zoomed imperfections.

Directional EC methods are not usually able to deal with multi-directional edges, which are involved in the missing block and this is one of the major problems. Li & Orchard (2002) offered an adaptive approach for a flexible edge direction finding within the block to recover a missing one with complex edge features. The corrupted blocks are restored by calculating a weighted linear combination (fourth order linear interpolation) of the neighbouring pixel values in eight directions. A sequential process is used and previously recovered pixels can be used in the recovery

afterward. Hence, this approach makes the interpolation more effective for restoring more detail information. Nevertheless, even though the PSNR result is improved, this method has a high level of complexity. Moreover, the restoration of details may not be accurate.

An enhancement for packet loss recovery especially for edge parts was put forward by Hsia (2004), which starts with a one-dimensional boundary search of neighbouring blocks to find the best match of the possible edge directions. Afterwards, the corrupted parts are estimated by a weighting linear interpolation using the estimated edge direction from the previous process level. Finally, the remaining missing blocks are calculated by a median filter.

In Zhao *et al.* (2005) a spatial error concealment algorithm was presented using directional extrapolation. The pixels within the corrupted block are recovered one by one. Each pixel in the corrupted block is recovered by two steps. The first step involves determining if there is an edge with one of ten directions traversing the pixel to be recovered. Then the pixel is recovered through simple extrapolation of two pixels along the determined direction. The experiment results showed that the proposed technique has achieved objective and subjective improvements compared with the previously published work.

An alternative to estimation of the missing block is a pixel-wise fine directional interpolation (FDI), which is based on the introduction of a spatial direction vector (SDV) (Kim, Koo & Jeong, 2006), (see Figure 2.14). SDV is calculated from the edge information of the surrounding available blocks by utilising a Sobel gradient filter and edge direction can be classified into one of eight possible directions (Figure 2.10). Then, the missing area is interpolated by best edge direction, based on the SDV vector and each pixel is estimated by the corresponding edge direction. The outcome shows an improvement over some of the other methods in the literature (Suh & Ho, 1997; Wang & Zhu, 1998; Salama, Shroff & Delp, 1998; Hsia, 2004) and can cope with complicated image structures.

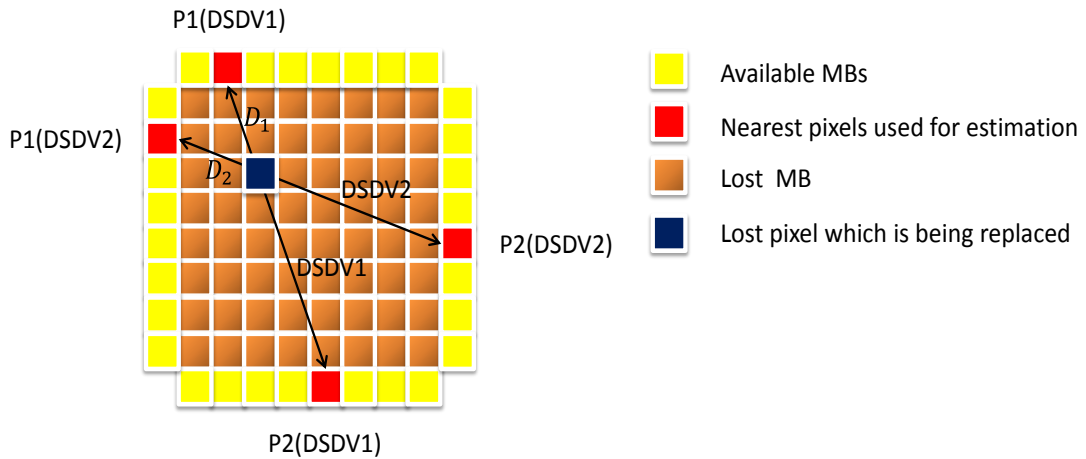


Figure 2.14: Restoration by dominant SDVs: DSDV1 and DSDV2.

Bayesian estimation is employed for EC to solve the problem of missing image blocks as a statistical machine-learning based interpolation (Zhai *et al.*, 2010; Liu *et al.*, 2014). The essence of the Bayesian philosophy applied to image restoration is to calculate the interpolation weights from the likelihood and prior functions, in turn calculated from the available information. Zhai *et al.* (2010) proposed a method which is a combination of a Bayesian framework and a DCT transform on a multi-scale EC platform. To begin, both missing and correctly received pixels are considered as vectors and then all missing as well as the pilot vectors gathered, are used to estimate the missing parts. In addition, the DCT transform, as mentioned before, helps to enhance the restoration. The image is first transformed to the DCT domain in a multi-scale frame and afterwards the missing details are restored through an iterative process, which takes account of previously recovered pixels as a guide for further improvement in the interpolation.

It can be seen from Figure 2.15, that the outcome of Zhai *et al.* (2010) is improved when compared with a number of state of the art methods. However, the result can be further improved to achieve a higher PSNR and better visual perception.

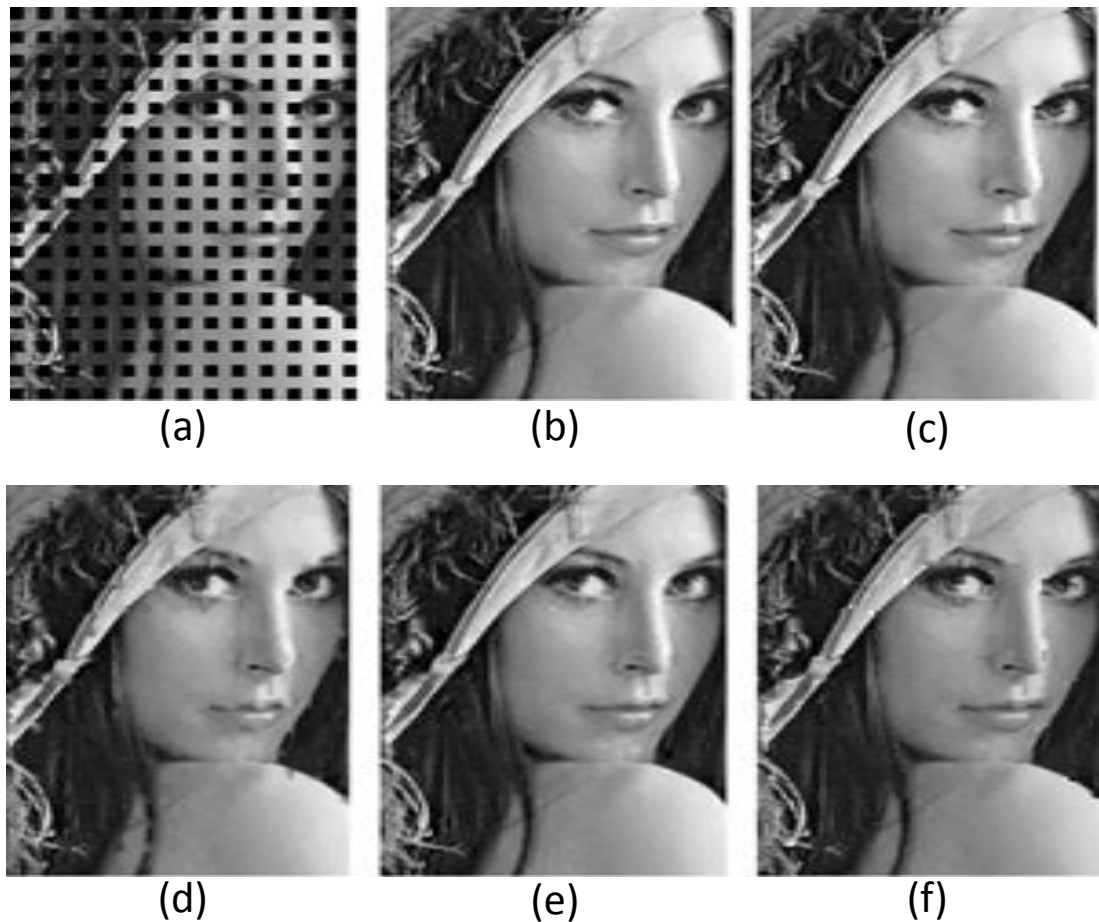


Figure 2.15: Comparison of different EC algorithms on Lena with 25% loss (8×8 isolated block loss). (a) Lossy image, (b) Bayesian EC (Zhai *et al.*, 2010), (c) BBF (Prochazka, Vysata & Jerhotova, 2010), (d) Sequential EC (Li & Orchard, 2002), (e) FSE (Meisinger & Kaup, 2004) and (f) POCS-based EC (Park *et al.*, 2005).

A further Bayesian estimation method based on an adaptive linear prediction by Liu *et al.* (2015) achieved a higher PSNR and better visual perception. Missing pixels are reconstructed sequentially, pixel by pixel, utilising linear prediction, with the order of the predictor being determined by adopting a Bayesian information criterion (BIC) and hence, more details and structures are restored and its PSNR is improved compared with the method of Zhai *et al.* (2010).

Another error concealment technique (Koloda *et al.*, 2013) is based on sparse linear predictions. The missing MB areas are restored sequentially using a linear predictor, the coefficients of which are estimated by an adaptive procedure based on sparsity and a missing data imputation approach. The estimation of the predictor coefficients is defined as a convex optimisation problem and then an alternative is derived based on an exponential approximation. Different exponential estimators can be utilised in

EC methods, such as in Zhai et al. (2008) Zhai *et al.* (2010) where a linear prediction model, a combination of sparse recovery and sequential filling, is proposed.

2.4.3 Combined Directional and Non-Directional Interpolation Methods

While directional interpolation methods are successful in recovering of the edges, they introduce stripe-shaped artefacts in the smooth parts of the image, such as edge-oriented directional interpolations that were investigated in Kwok & Sun (1993) and Hsia (2004).

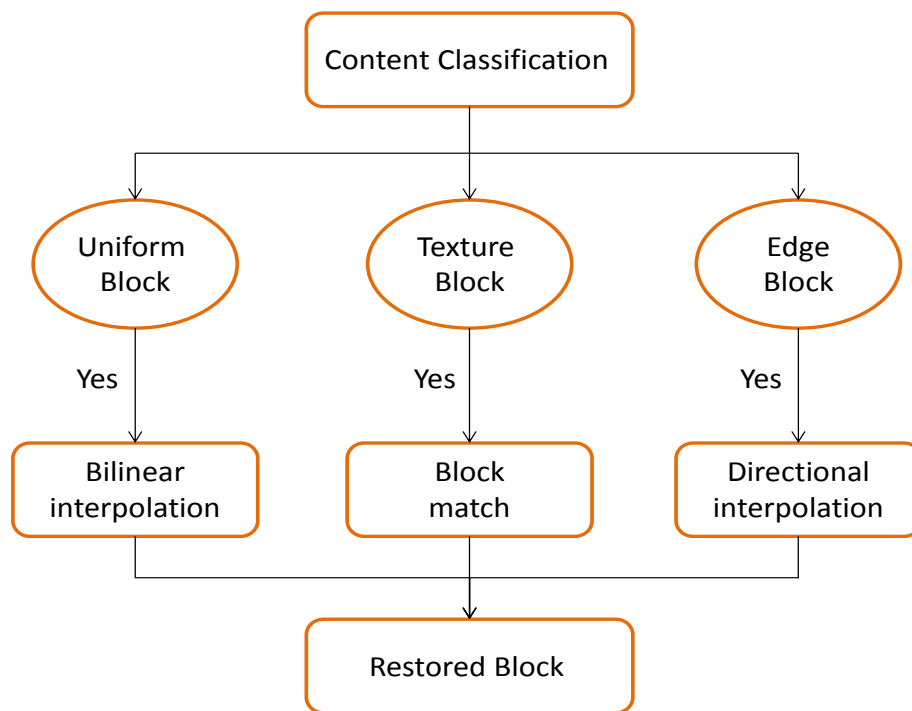


Figure 2.16: Framework of content adaptive spatial error concealment.

In Rongfu, Yuanhua & Xiaodong (2004) an adaptive method was proposed in order to develop an EC algorithm that benefits from the best of a combination of directional and non-directional methods. Two steps are involved in this technique. First, the type of the error block (EB) is detected and classified into one of the following types: uniform, texture or edge groups. Then, a suitable EC method is applied to each category to conceal the missing block. As illustrated in Figure 2.16, bilinear interpolation, best block matching and directional interpolation are used for block of type uniform, texture and edge categories, respectively. For this method and in general for all adaptive methods, image blocks need to be categorised correctly to

achieve high performance, for in cases of wrong classification the result will be severely reduced quality.

2.5 Conclusion

There are different causes for image loss in broadcast networks; however, the result is degradation of the image quality. Many interpolation methods have been proposed for estimating the missing parts, when there is a loss in the received image, with the aim of recreating an acceptable image quality, but problems still exist. The main ones associated with these interpolation methods relate to the inability to recover edges and details correctly. In particular, a reliable and effective interpolation method is necessary for hiding the effect of missing blocks in still and moving images.

Having reviewed some of the most relevant literature, the EC interpolation approach is considered to offer the most fruitful avenue of enquiry and hence, is chosen as the focus of the current research. Spatial reconstruction techniques take various forms, but all are based on the assumption that there is high level of correlation between neighbouring macro blocks. There are some key elements required for a reliable EC method:

- The EC method needs to extract accurately the statistical correlation of the neighbouring blocks;
- The EC method needs to extract effectively the spatial structure of the neighbouring blocks including the edges;
- The EC method must be robust;
- The EC method must be computationally efficient.

It has been observed from the literature review that it is essential to design a new more effective error concealment method to improve upon the existing ones and thus, achieve a suitable reconstruction quality for the corrupted image. The new approach needs to cover all types of block including, smooth, edge and texture. To this end, for this research an effective method based on both local/global edges analysis and DCT/DWT transform in a multi scale skeleton is proposed.

For this research, image restoration is investigated within a pyramid structure that lends itself to the use of DCT or various families of DWT as the kernel function. The main justification for the choice of a pyramid as the framework for interpolation is pragmatic: a relatively large gap at the base is reduced to a single sample at the apex which can be conveniently interpolated. The result can then be progressed downwards towards the pyramid base through a set of repeated and scaled-up operations.

Chapter 3

Image Processing Methods and Tools

Chapter 3

3. Image Processing Methods and Tools

3.1 Introduction

This chapter provides an overview of the theory and applications of image processing methods and tools used in this thesis. The focus is on 1) image processing basics, 2) discrete cosine transforms (DCT), 3) wavelet transforms and 4) edge detection methods.

3.2 Image Processing History

The history of digital images can be traced back to 1920s when the first ones were sent through the submarine cable (Bartlane cable) telegraph between London and New York, which were used in newspaper publication (Gonzales & Woods, 2008). An early example of the application of digital image processing was in July 1960 when the first pictures of the moon were transferred from the Ranger 7 to the Jet Propulsion Laboratory and enhanced by a computer to improve the quality (Banham & Katsaggelos, 1997).

In addition to the space programme, many other digital image processing applications were developed between 1960 and 1970. The image processing field has evolved over the last half century to encompass a broad range of everyday applications in: astronomy, medicine, physics, automation, security, biometrics, television, social media and entertainment.

For example medical imaging called computerised axial tomography (CAT or CT for short), developed in 1970, is one of the main medical applications in image processing, using X-rays for medical diagnosis (Gonzales & Woods, 2008).

In order to understand image processing applications, first, it is necessary to comprehend what a digital image really is. Generally, an analogue image is captured by one of the imaging machine devices, whereas a digital image is obtained by applying a sampling and quantisation process to the real natural image (Schönlieb, n.d.).

3.3 Image Processing Basics

Although human vision is limited to the visual band of the electromagnetic spectrum, imaging machines can cover the entire spectrum from sound to gamma waves, such as ultra sound. One of the easiest ways to have a fundamental understanding of image processing is to categorise images by their sources of energy, which can be any of the following: electromagnetic stream, ultra sound, electronic or computer based. Nearly all images of scenes originate in the reflection/absorption of an illumination source.

3.3.1 Image Acquisition

Figure 3.1 shows how illumination from the source of energy is transformed into a voltage waveform. Sensor elements are sensitive to a specific type of energy and when the illumination reaches the sensor after passing a filter, each element reacts to the received energy. The combination of the input power and reaction of the sensor to the energy produces a continuous output voltage, as can be seen in the Figure 3.1. Usually, the nature (amplitude and spatial) of the output voltage is based on the physical information of the source energy (Curtin, n.d.).

There are different types and configurations of sensors, from a single element to a line or array, which are categorised based on the structure of the sensor elements. In order to create a digital image the continuous voltage outcome needs to be transformed into digital information. Sampling and quantisation are two processes required to convert the continuous sensed data into a digital format.

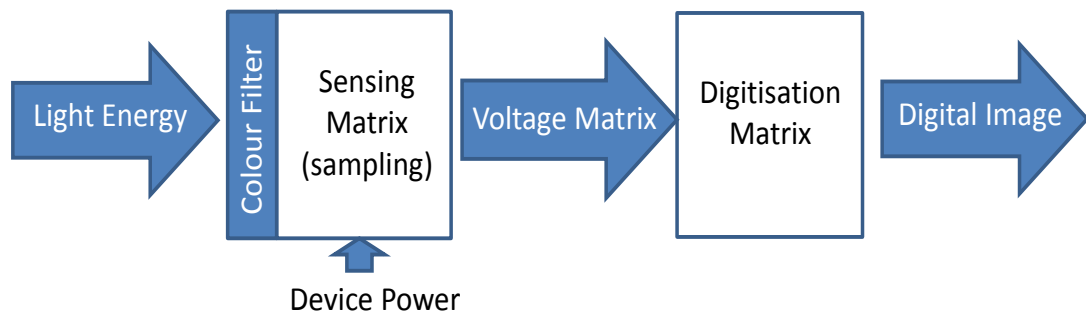


Figure 3.1: Input-output block diagram of an imaging sensor. Input: light energy (photons), output: sampled and digitised voltage.

3.3.2 Image Sampling and Quantisation

As explained in subsection 3.3.1, there are various methods to acquire images, with their outcomes being a continuous (in both spatial and amplitude) voltage waveform, which is nearly the same for all of the techniques. Sampling and quantisation are the two main processes for converting the output of sensing machines into the digital images in spatial and amplitude forms, respectively. Sampling refers to the process of digitising the coordinates of the image, whereas quantisation pertains to digitising of the amplitude.

The digitisation process starts with sampling the image into separate lines of elements, which include the continuous sample of the image by taking a line in the horizontal direction and then, taking the samples from that line at equal intervals (Figure 3.2.a). From this, the discrete values of the image coordinates are obtained.

Then, in order to provide discrete values for the intensity of each sample, usually a scale of discrete intervals is used for quantisation. The scale can include various values of intervals ranging from white to black, and then a specific amount is given to each sample depending on the pixel intensity amplitude (Gonzales & Woods, 2008).

Figure 3.2 shows the image that has been captured by the imaging device and a line $c - d$ has been set to provide a continuous sample. The continuous line is converted into the discrete samples by selecting separate elements in sufficiently small equal intervals (sampling interval $< 1/(2 \times \text{Bandwidth})$), as can be seen in Figure 3.2 (b).

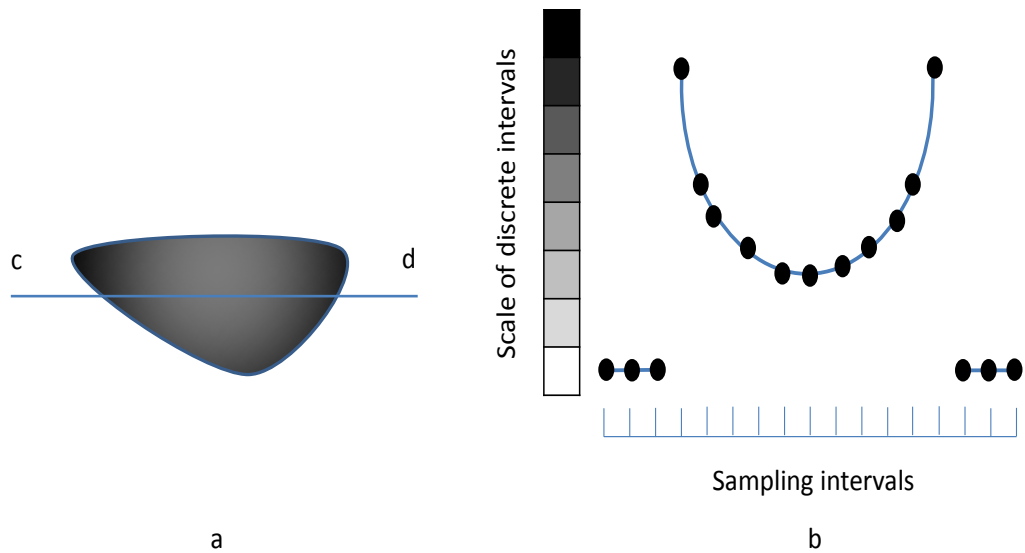


Figure 3.2: a) continuous image captured by the imaging machine, b) sampling and quantisation of the captured image.

The intensity value of each sample is quantised according to a discrete interval scale, and in this example the amplitude scale includes eight values. Figure 3.3 (a) shows the same image on a sensor array and the result of digitising that image is illustrated in the Figure 3.3 (b).

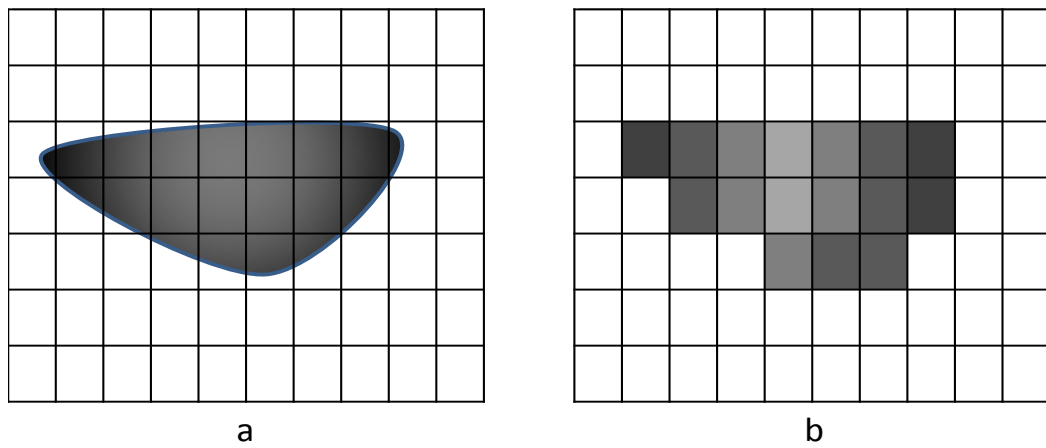


Figure 3.3: a) sensor array with a continuous image, b) sampled and quantized image.

3.3.3 Digital Image Representation

A digital image is a grid of elements (values) called pixels (Figure 3.4), of which there are two types: grey valued and colour valued. The former consist of scalar

values ranging from between 0 (black) and 255 (white), whilst the later include three vector values (r, g, b). Each individual pixel in a colour image consists of three separate primary colour intensities, red, green and blue (RGB), as a multi-valued function, which combine together to produce the whole image.

The representation of the image in mathematical form is called the image function $A(x, y)$ and is a two dimensional function in the x and y directions with an amplitude of A . The image pixels values can be represented as a matrix (Figure 3.4).

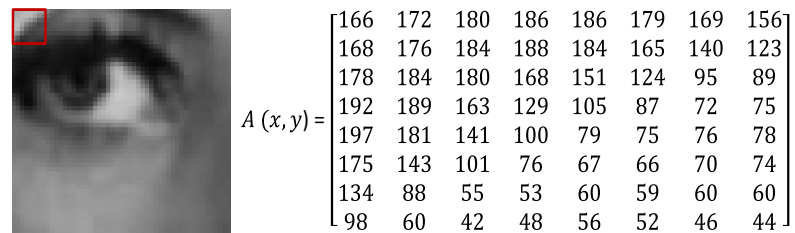


Figure 3.4: A part of Lena image showing the image pixels. Matrix of 8×8 pixel values which is marked by a red square in the image.

A true colour specification for an image file requires an m-by-n-by-3 array of RGB values. The first, second and third matrix contains the red, the green and the blue components of each element, respectively, in the image (Figure 3.5).



Figure 3.5: Lena.BMP 512×512 pixels, 768 KB and the individual RGB components.

The image degree of brightness is quantised into a sample format, a common form of which is 8-bit integers, but this can only represent $2^8 = 256$ discrete values (Kolås, n.d.). This means that 2,097,152 bits would be needed to represent the entire 512×512 image with 256 grey levels. Thus, reducing the content of the data for storing, transmitting and downloading while preserving the quality is very important

in image processing, for which various types of image formats have been proposed, with some of these being covered briefly in the next subsection.

Resolution in an image is the amount of detail it contains, and directly depends on the size of the sampling intervals that the pixels represent in the image. Greater detail and smaller pixel size leads to higher resolution. Even though the resolution range available varies, nowadays, there are much higher resolution images than before, such as that of the Machu Picchu, which is claimed to be the highest resolution image ever taken. Jeff Cremer is the photographer and the resolution of the image, which was taken with the Canon 7D and a 400mm lens, is about 15.9 giga-pixels (Machu Picchu, n.d.).

3.3.4 Image File Formats and Image Compression Algorithms

After acquisition and digitising, an image is composed of pixels, which include 24 bits of resolution that define its colour. An image should be in one of a number of standard file formats in order to be displayable on a mobile phone, camera, computer or printer. The file format is essentially the compressed and encoded form of the raw digitised image. The different formats differ in terms of the method of compression and encoding of the digital samples and the quality and the size of the output file. In addition, there are different types of image files, including vector, compressed and uncompressed formats.

Compression for binary data is completely different from the algorithms that have been proposed for an image compression, and this is because of the specific statistical features of the image. The main aim of image compressing techniques is to save storage area and bandwidth as much as possible while losing little or no visible quality. Image compression also results in spending less time and power in transferring and receiving an image as well as providing faster results in search engines.

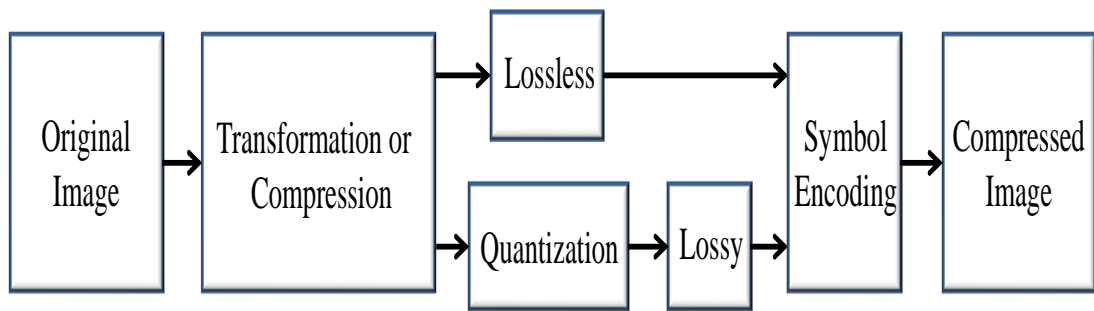


Figure 3.6: Compression framework of both the lossy and lossless methods.

In general, image compression methods can be divided into two main categories (Figure 3.6). The first, the lossless method, exploits the data redundancies to reduce the size of the file, but keeps an exact replica of the original data and this is usually used for archival purposes, such as documents, medical records or art related images, where having a copy of the original image is necessary (Rehman, Sharif & Raza, 2014).

The second category, the lossy technique, is often used in the situation where some, usually imperceptible, loss is acceptable in order to reduce the bit rate, such as for digital TV, video and photographs in applications. Usually, the output file size of lossy methods is less than with the lossless techniques for the same image (Rehman, Sharif & Raza, 2014).

a) Lossless formats

BMP (bitmap): This is an uncompressed format which is used by the Microsoft Windows graphics device interface (GDI) subsystem and operating system (OS/2). The bitmapped graphics format normally takes the form of a simple graphics file, from which image dimensions and different colour depths to show the resolution can be obtained.

PNG (Portable Network Graphics): This format is a lossless data compression proposed by the Internet Engineering Steering Group in 1996. As this method was developed for the Internet, it is not suitable for high-quality print graphics and as a result, cannot be used in non-RGB colour spaces. Moreover, the PNG type is suitable for images with small variation in colour or large areas of the same colour.

TIFF (Tagged Image File Format): This method was proposed by the company Aldus in the mid-1980s, in order to have a common file format for image scanning across all scanner companies. At that time, it had just two binary values for each pixel, but after progress was made in scanner quality, it could handle both grey and RGB images.

In general, lossless compression, such as PNG and TIFF, cannot be applied to high quality images that have very bright colours and textures. In scenarios when an image has great detail lossless algorithms do not work well in compression, working better with those images that have small variation in colour or large areas of the same colour.

b) Lossy formats

JPEG: The acronym JPEG refers to the Joint Photographic Expert Group, which proposed the method in 1992 and this technique can work for both colour images and grey scale ones. The rate of compression that can be selected by users starts from 100 to 1 with the higher the number giving less compression and better quality. Consequently, there is an inverse relation between the image quality and the storage size, whereby picture-quality can be traded for a smaller file size.

For compression, JPEG uses the discrete cosine transform, which converts all sources from the spatial domain into the frequency domain. Then, it removes the high frequency parts by quantisation, thus only keeping the lower frequency part. Following this, a sequence of quantised factors is packed in a bit stream ready for transmission. This method is generated based on the human visual system, which is more sensitive on parts that are alike in order to identify those that have more variation with high frequency and also, luminance is considered more important than colour.

JPEG format is ideal for large images because it has a large compression ratio which reduces the image quality. As can be seen from Figure 3.5 and Figure 3.7 the size of the Lena image (512×512) file is reduced from 768 KB in the BMP format to 61 KB in the JPEG format. But on the other hand, the JPEG compression format is not suitable for images that require the original copy of the data to be reconstructed and

in addition, it is not good for textual or iconic graphics, because noticeable artifacts can occur due the sharp contrast between adjacent pixels. Furthermore, this format is not suitable for scenarios where the image might be edited several times, because in this case it loses its quality (jpeg. n.d.).



Figure 3.7: Lena.JPEG 512x512 pixels, 61 KB and its individual RGB components Lena.JEPG.

JPEG2000: This is a wavelet-based compression method which was also proposed by the Joint Photographic Experts Group committee in 2000 (Boliek, Christopoulos & Majani, 2000). It was developed to meet the demand for a new standard to apply as a compression technique in different still images types and is capable of providing effective lossy and lossless compression. JPEG2000 is optimised for efficient, scalable, and interoperable compression (Skodras, Christopoulos & Ebrahimi, 2001).

The wavelet transform is used in JPEG2000 as an alternative of DCT in JPEG. Figure 3.8 shows that the wavelet is first applied to the original image, then the coefficients are quantised and consequently, the compressed image can be produced by using Huffman coding to encode the coefficients.

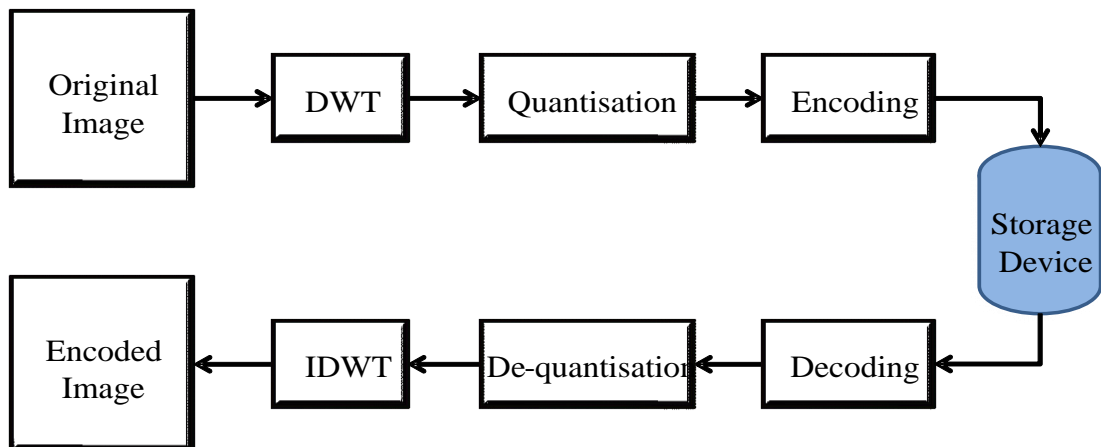


Figure 3.8: Wavelet-based compression method.

3.3.5 Packet loss

A number of image blocks are combined together into a packet and then transmitted in packet based networks, each packet being transmitted separately with its own address. However, the address field can be lost due to network congestion or signal fading, which leads to packet loss. Consequently, during the transmission of a DCT compressed image bit errors or packet loss can be produced by compression or network congestion, respectively. The result of this problem manifests itself as bit error, packet delay, packet loss, packet intrusion or block de-synchronisation. Some samples of the user's visual experience being distorted with corrupted Lena are shown in Figure 3.9.

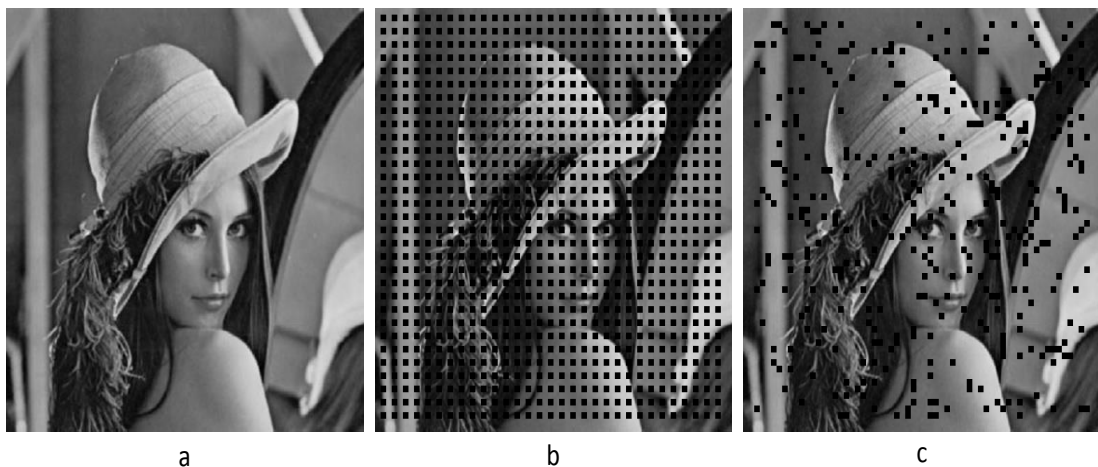


Figure 3.9: a) Original Lena image, b) Lena image with a 25% regular 8×8 missing macro block and c) Lena image with a 10% random 8×8 missing macro block.

3.4 Discrete Cosine Transform (DCT)

The discrete cosine transform (DCT) was proposed by Ahmed, Natarajan & Rao (1974) for use in various digital image processing areas. By using DCT, the data can be transferred from the time or spatial domain to the frequency domain and represented by an efficient reduced set of frequency coefficients (Haberdar, 2012).

DCT is a way of encoding the original data with a new set of cosine basis functions and the representation of information in this transform is a set of cosine functions at a range of frequencies from zero to half the sampling rate (Figure 3.11).

DCT is now the mainstream transform of choice for many popular commercial coding applications, particularly in image processing, but also in audio processing. It has been applied in various well known international applications for image/video storage and transmission, such as in JPEG (jpeg. n.d.), the Moving Pictures Expert Group (MPEG) (Aign & Fazel, 1995), MP3 and the International Telecommunications Union's (ITU) recommendation for H.261 and H.263 (Wiegand, *et al.*, 2003).

For image processing, DCT has the important property that the images are compressed to a relatively small region of the low frequency coefficients centred on and around the zero frequency, which itself represents the mean image value. However, some higher frequency coefficients are needed in images that contain significant edge information. Nevertheless, for most images retaining only 16% of the lowest image coefficients can provide a relatively good reproduction of the image (Klein, 1990).

DCT is a close approximation to the theoretically optimal (for Gaussian signals) Karhunen–Loève (KL) transform (Dony, 2001) or principal component (PC) analysis (Pauluzzi & Beaulieu, 2000). KL and PC transformation incur an additional computational cost in estimation of the covariance matrix (or tensor, 3-D matrix) of the image from which the transform matrix is calculated. In contrast, DCT's basis functions are fixed, but still provide a good approximation, in terms of efficiency of performance, when compared with the KL and PC transformations.

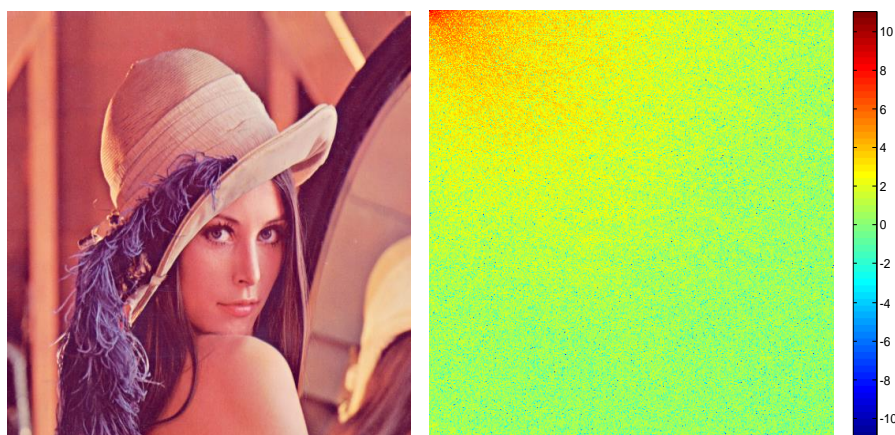


Figure 3.10: Original colour Lena and the energy compression map of this image after applying a large 512×512 DCT.

By using orthogonal cosine transforms, the number of significant coefficients is relatively small and hence, other less significant ones can be eliminated. Thus, a significant reduction in computation load is provided at the cost of a small gain in the mean-square estimation error when compared with state of the art transforms, such as fast Fourier and Haar (Ahmed, Natarajan & Rao, 1974).

Nowadays, one of the most popular and effective transform coding algorithms is DCT coding, which is able to compress most of the signal energy within a few coefficients and the degree of compression depends on the correlation structure of the data (Haberdar, 2012). For perceptually good reconstruction only a few high energy low frequency coefficients are required. In addition, usually a DCT is used to restore image corruption with macro-blocks of 8×8 .

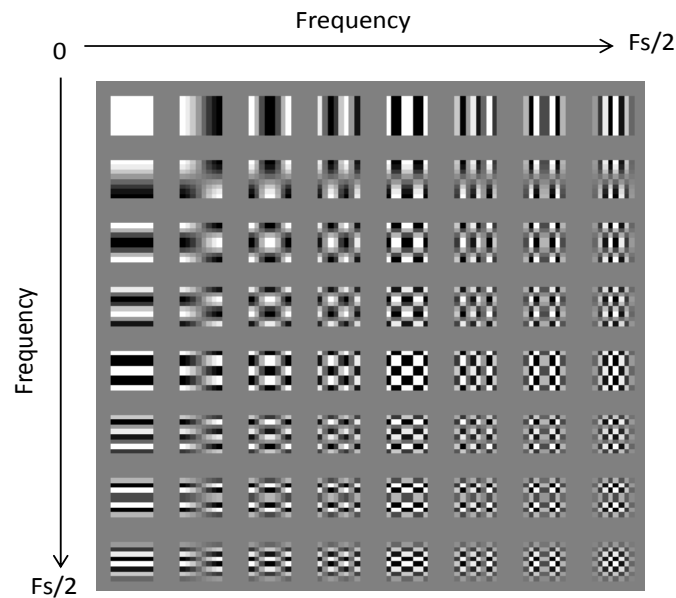


Figure 3.11: Illustration of 64 basis functions of 8×8 2D DCT. Top left corner shows lowest (zero) frequency and bottom right corner the highest. Note, energy colour coded: white highest energy and black zero energy.

pixels and the intensity transitions between these blocks become more and more apparent when the high-frequency data are eliminated due to heavy quantisation (Belfiore *et al.*, 2003).

3.4.1 DCT Equations and Basis Functions

a) One-Dimensional DCT

The general equation for a 1-D (M data samples) DCT applied to a signal x is defined as:

$$F(p) = \left(\frac{2}{M}\right)^{\frac{1}{2}} \sum_{i=0}^{M-1} \Lambda(i) \cos\left(\frac{\pi p}{2M} (2i + 1)\right) x(i) \quad (3.1)$$

Note, successive basis functions have a frequency spacing of $\frac{\pi}{M}$ or equivalently $\frac{F_s}{2M}$.

In matrix form this equation can be expanded as:

$$F = TX \quad (3.2)$$

or:

$$\begin{pmatrix} F_0 \\ F_1 \\ F_2 \\ \vdots \\ F_{M-2} \\ F_{M-1} \end{pmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & \cos \frac{\pi}{2M} & \cos \frac{3\pi}{2M} & \cdots & \cos \frac{\pi(2M-1)}{2M} \\ 1 & & & & \cos \frac{\pi(2M-1)}{M} \\ 1 & \vdots & \vdots & \vdots & \vdots \\ 1 & & & & \cos \frac{\pi(M-2)(2M-1)}{2M} \\ 1 & & & & \cos \frac{\pi(M-1)(2M-1)}{2M} \end{bmatrix} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_{M-2} \\ x_{M-1} \end{pmatrix} \quad (3.3)$$

where, the elements of matrix T are the DCT basis functions, M is data samples, p is an entry of the DCT of an image and x is an entry signal.

DCT is basically the same as DFT, but the difference is that the former only uses the real part of the equation and not the imaginary part.

DCT is a block-based algorithm that divides an image into separate $M \times N$ blocks (usually macro blocks of 8×8 pixels) and transfers them separately from the $M \times N$ samples in the spatial domain to $M \times N$ samples in the frequency domain. The frequency domain resolution is $\frac{F_s/2}{M}$.

b) Two-Dimensional DCT (DCT2)

Two-dimensional DCT takes a block of $M \times N$ coefficients from the image and then following transformation in the DCT domain, most of the energy is concentrated in the top left low-frequency corner, whereas the higher frequency components have relatively lower energy and are less significant. Thus, after applying DCT, data reduction or bit-resource reduction, can be achieved by concentrating more (bit) resources on the few highest value coefficients. This strategy is able to provide compression of the data with consequent memory/power/bandwidth/computation savings.

For example, because the human eye is less sensitive to low energy higher frequency coefficients, if the coefficients over a certain threshold are discarded, this will not have a significant impact on the visual result.

A flow chart of a general application of DCT in image processing is illustrated in Figure 3.12.

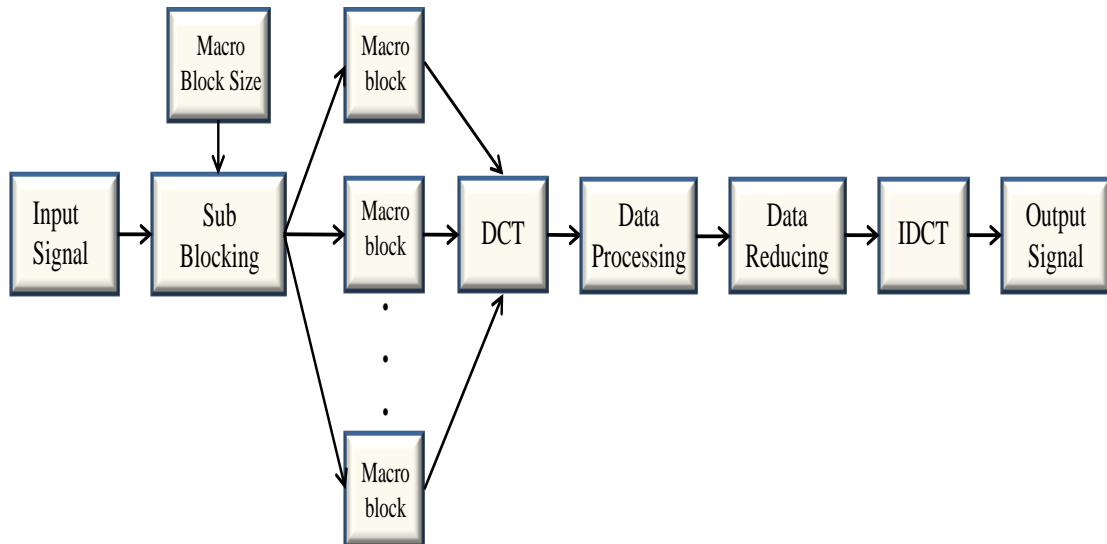


Figure 3.12: A image processing flow chart incorporating DCT as the main transform for data representation and data reduction.

For image processing, DCT is applied across all rows/columns of the image and the values are calculated using the DCT formula. The 64 (8×8) DCT basis functions are illustrated in Figure 3.11 and it can be seen that the higher energy is concentrated in the top left corner.

The 2D-DCT of an $M \times N$ matrix F is defined as:

$$F_{p,q} = \left(\frac{2}{M}\right)^{\frac{1}{2}} \left(\frac{2}{N}\right)^{\frac{1}{2}} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} x(i)x(j) \cos\left(\frac{\pi p}{2M}(2i+1)\right) \cos\left(\frac{\pi q}{2N}(2j+1)\right) f(i,j) \quad (3.4)$$

where, Equation 3.4 computes the p, q entry of the DCT of an image, M and N represent the size of the DCT block, $f(i, j)$ is the p, q element of the input image which is represented by matrix A .

A commonly used macro block size in applications of DCT to image processing is 8×8 and this is also the case in the proposed methods employed in this thesis $M = N = 8$, and the indices p, q are in the range from 0 to 7. Figure 3.13 shows the DCT of the Lena in four down-sampling levels.



Figure 3.13: From left to right, DCT of Lena image and sub-images of size: (512×512) , (256×256) , (128×128) and (64×64) pixels, respectively.

The inverse discrete cosine transform (IDCT) is a process for reconstructing the image (Equations 3.5 and 3.6).

$$f_{i,j} = F^{-1}_{p,q} = IDCT2(F_{p,q}) \quad (3.5)$$

$$F^{-1}_{i,j} = \left(\frac{2}{M}\right)^{\frac{1}{2}} \left(\frac{2}{N}\right)^{\frac{1}{2}} \sum_{p=0}^{M-1} \sum_{q=0}^{N-1} x(p)x(q) \cos\left(\frac{\pi p}{2M}(2p+1)\right) \cos\left(\frac{\pi q}{2N}(2q+1)\right) F(p, q) \quad (3.6)$$

where,

$$x(\varepsilon) = \begin{cases} \frac{1}{\sqrt{2}} & \text{if } \varepsilon = 0, \\ 1 & \text{otherwise.} \end{cases}$$

By utilising Equation 3.6, the most important DCT coefficients can reconstruct the image effectively by using the zero padding method.

As an alternative representation, the output DCT vector is computed by applying the DCT Matrix T_{pq} on the matrix A input vector. The DCT Matrix T_{pq} can be expressed as (Equation 3.7):

$$T_{pq} = \begin{cases} \frac{1}{\sqrt{N}} & \text{if } p, q = 0 \\ \sqrt{\frac{2}{N}} \cos \left[\frac{(2p+1)q\pi}{2N} \right] & \text{if } p, q > 0 \end{cases} \quad (3.7)$$

The result of $T * A$ is a matrix with columns that are the one dimensional DCT of the columns of A and Equation $B = T * A * T'$ represents the two-dimensional DCT. The inverse of matrix, T' , can be calculated by transposing the original matrix, because T is an orthogonal matrix.



Figure 3.14: Left, original Boat image and right, the reconstructed image using only 7% of the low-frequency 2D-DCT coefficients and applying 2D-IDCT (8×8 macro blocks).

Figure 3.14 shows reconstruction using 2D-DCT and 2D-IDCT of (8×8) blocks in the Boat image, with the number of coefficients set to zero in the DCT matrix being 58 out of the total of 64 in each macro block. Even though 93% of the image DCT

coefficients are eliminated and hence, the size of the image file is greatly decreased, the image is still visually in-differentiable from the original. In general, the quality of the image can be improved by increasing the number of DCT coefficients used in the reconstruction.

3.4.2 DCT for Resizing Images

Zero-padding of a signal followed by a frequency transformation is a method of interpolation or up-sampling in time/space or frequency domains, which can be used in two ways:

- (1) Zero-padding a signal in time/space followed by a frequency transformation results in interpolation in spectrum; it yields a spectrum with additional interpolated spectral lines, thus achieving a higher ‘apparent’ frequency resolution.

$$[N \text{ signal samples, } N\text{-zeros}] \xrightarrow{FT} [2N \text{ signal samples}]$$

- (2) Zero-padding in the frequency domain followed by an inverse frequency transform provides interpolation in time/space.

$$[2N \text{ signal samples}] \xleftarrow{IFT} [N \text{ spectral samples, } N\text{-zeros}]$$

Zero-padding combined with DCT can be used to resample and resize an image. Shirani, Gallant & Kossentini (2001) proposed zero padding in order to control the redundancy for images in the DCT domain. Zero-padding facilitates oversampling by adding a number of zeros to the transformed image, whereby some redundant information is combined with the original data in the frequency domain. Thus, when first applying DCT additional zeros are padded into the image and then the IDCT process is applied on the larger signal.

The process of up-sampling using DCT is illustrated in Figure 3.15. To up-sample an image from a size of $n \times n$ to a size of $p \times p$, first, a 2D DCT is applied on the rows and then to the columns of the input image with a size of $n \times n$ pixels. This yields an $n \times n$ DCT matrix, which is augmented to size $p \times p$ by appending $p - n$ rows and

columns of zeros. Consequently, the number of pixels (image size) is increased by a factor of (Tillo & Olmo, 2007):

$$v_p = (p^2 - n^2)/n^2 \quad (3.8)$$

Then, the zero-padded DCT matrix is transformed to the image domain by an IDCT, which yields an up-sized image with dimension $p \times p$.

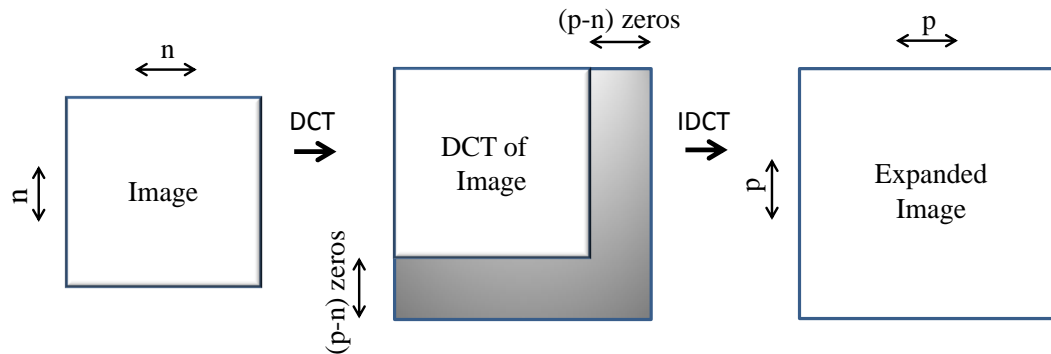


Figure 3.15: Image size expansion: zero-padding in the transform domain to oversample or reconstruct a truncated input image.

Figure 3.16 shows a successive, two stage, application of up-sampling via DCT to an image of boats and lighthouse. The oversampled image is in fact the original image at a higher sampling rate. However, it must be noted that the actual image resolution is not increased since the new data have only been interpolated and therefore, the pixel samples are less close to each other with their values being more correlated. The amount of the padding determines the oversampling factor and hence, the level of redundancy introduced in the image.

3.4.3 Zig-zag Scanning

Zig-zag scanning is a method of scanning a matrix or image into a vector. For DCT it allows the DC coefficient (the coefficient with zero frequency in both dimensions) and the lower frequency AC coefficients (remaining 63 coefficients with non-zero frequencies) to be scanned first. With this method, the coefficients are scanned in a zig-zag order to convert the 2-D DCT coefficients into a 1-D data stream. For example, it can map a 8×8 macroblock to a 1×64 vector and gather low frequency coefficients at the top of the vector.

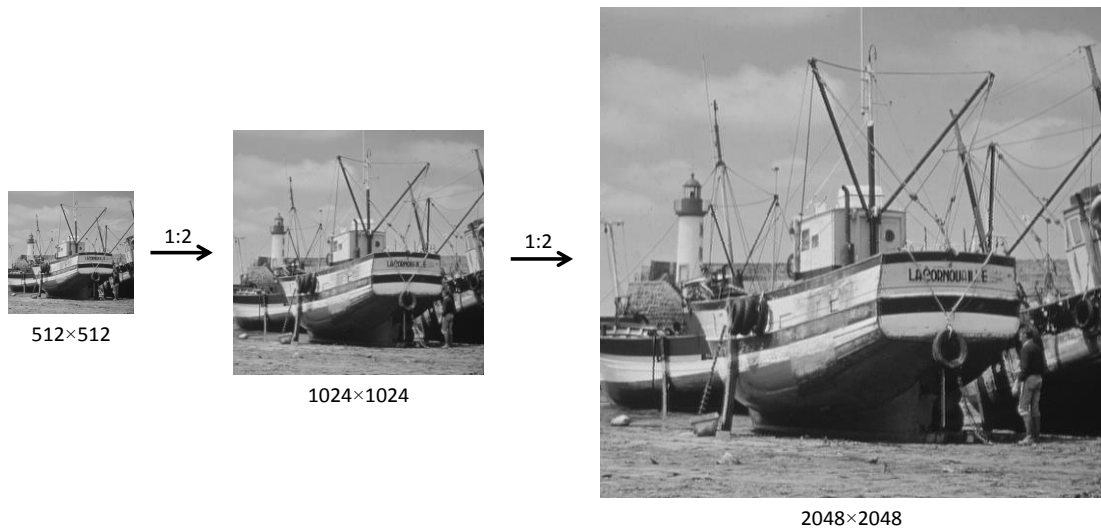


Figure 3.16: Actual application to images 8×8 MBs (512×512) transformed to 16×16 (1024×1024) or 32×32 (2048×2048); useful for fitting to screen displays of different sizes.

In general, the frequency is increasing in a DCT matrix in the first dimension and is also increasing in a column in the second dimension. Since after zig-zag scanning the AC coefficients with larger energy usually locate at the first few entries in the 1-D data stream, then higher coding efficiency can be achieved (Ding, Wei & Chen, 2011). Figure 3.17 shows the process of zig-zag scanning.

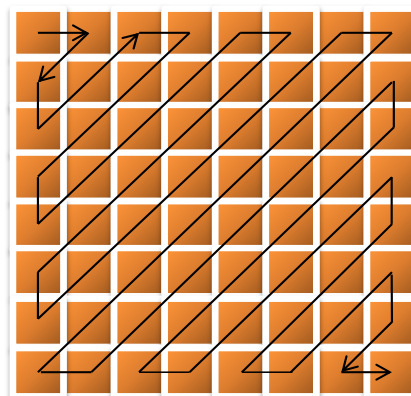


Figure 3.17: Zig-zag scanning for an MB (8×8) in 2D-DCT.

3.5 Discrete Wavelet Transforms

Wavelet transforms are commonly used as alternatives to the more historically established Fourier transform (FT), which was proposed in 1950 and the DCT (as explained in section 3.3).

Conventional frequency transform methods assume that the input signal is stationary (i.e. time/space invariant) and employ a set of stationary waves, such as cosine and sine functions of different frequencies as the basis functions for the transformation of a signal. The term stationary implies that the signal's statistical features, such as the mean value, power/variance and higher order statistics remain constant. A sinewave is an example of a stationary signal, whereas a bird chirp is an example of a non-stationary one.

Problems arise due to the non-stationary time/space-varying nature of signals such as audio and image. The solution is short-time (block) frequency transforms where the signals are divided into short segments, such that within each segment the signal can be considered as relatively stationary.

Wavelets employ non-stationary basis functions and, at least in theory, are better suited to representation of non-stationary signals, such as speech, music or image. The basis functions of wavelets are transient signals of different frequencies and scales that are often derived from the impulse responses of a pair of mirror-imaged low-pass and high-pass 'mother' wavelets (Gonzales & Woods, 2008).

The mathematician Alfred Haar is credited as the first person to put forward the idea of wavelets in 1909. Subsequently, the concept was reintroduced in the context of modern digital image processing, by Jean Morlet in 1981. Morlet continued his research with Alex Grossman and in 1984, they coined the term of wavelet (Liu, 2010). The first orthogonal wavelets were proposed by Stromberg in the early 1980's, whilst the second type was introduced by Yves Meyer in 1985 (Jawerth & Sweldens, 1993).

Multi-resolution is one of the important aspects in the wavelet proposed by Meyer and Stephen Mallat in 1988. Subsequently, Ingrid Daubechies contributed a systematical method for the development of a compact support orthogonal wavelet. Moreover, a fast wavelet transform was proposed by Mallat in 1989, which was one of the first examples of wavelet applications in the image processing field (Liu, 2010).

3.5.1 Wavelet Analysis-Synthesis

Wavelet transformation is one of the main techniques for time-frequency signal transformations. Wavelets are made of square-integrable (finite energy) functions, and can include real/complex values that are generated by a wavelet using specific orthonormal series (Walker, 2008). Generally, wavelets are short wavelike functions that can be scaled (dilated) and translated (shifted), whereby in a wavelet transform each signal can be represented as a scaled and translated wavelet.

A discrete wavelet transform (DWT) is used in this work, because it can analyse the signal with various resolutions at different frequency bands by a simple procedure, as it decomposes/separates the original signal into two parts, approximation and details, thus allowing independent analysis of the coefficients at different scales (Khaziakhmetov & Zakharova, 2012).

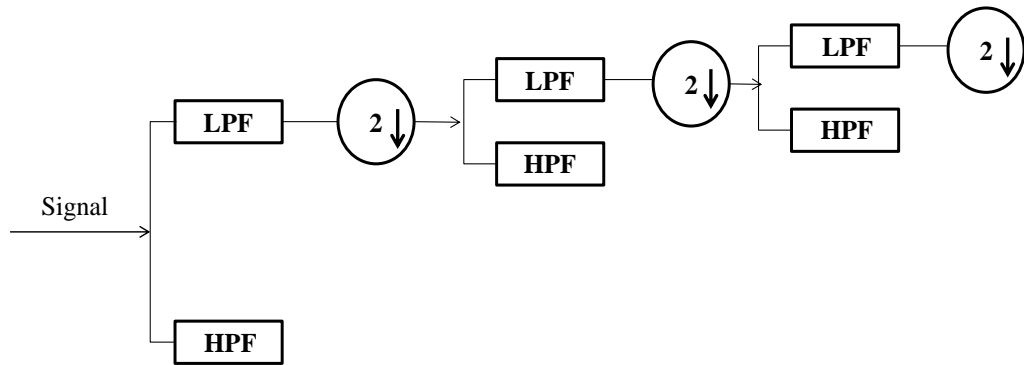


Figure 3.18: Schematic diagram of a 1D wavelet transform.

Wavelet decomposition is affected by a cascade series of image processing blocks, with each block consisting of a low-pass filter, a high-pass filter and down samplers, as depicted in Figure 3.18. The input-output relationship of the combined filtering and down-sampling operation can be expressed as:

$$y[n] = \sum_{k=-\infty}^{\infty} h[k].x[2n - k] \quad (3.9)$$

where, $h[k]$ are the filter coefficients and $2n$ signifies down-sampling. Successive high-pass and low-pass filtering is applied to the time domain signal in order to decompose it into different frequency bands. Firstly, a half-band high-pass filter

(HPF) $g[n]$ is applied to the original signal $x[n]$, and then the outcome is passed through a low-pass filter (LPF) $h[n]$. After the filtering, half of the samples can be eliminated (down-sampled) according to the Nyquist's rule, whereby the signal can be subsampled by factor of 2, simply by discarding every other sample (Polikar, 2006). This decomposition process can be repeated many times and produces multi-resolution layers, which constitute one level of decomposition and can be mathematically expressed as follows:

$$y_{high}[n] = \sum_n g[k].x[2n - k] \quad (3.10)$$

$$y_{low}[n] = \sum_n h[k].x[2n - k] \quad (3.11)$$

where, the results of the high-pass and low-pass filters after decomposition by a factor of 2 are $y_{high}[n]$ and $y_{low}[n]$, respectively. The block diagram of a 1D DWT transform can be seen in Figure 3.18.

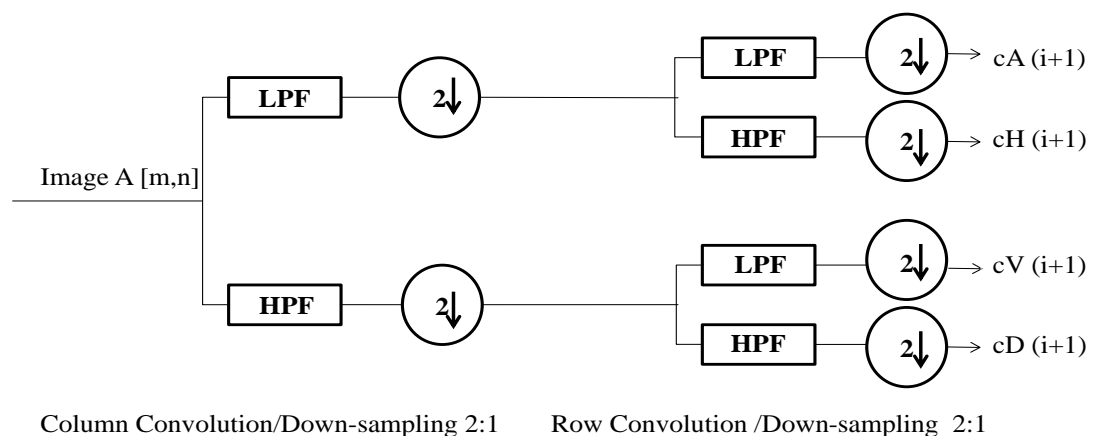


Figure 3.19: Schematic diagram of sub-band decomposition for one level of a 2D wavelet transform.

For image processing applications we need a wavelet that is two-dimensional (Figure 3.19). Generally, two one-dimensional wavelets or scaling functions are combined to produce a two-dimensional wavelet (Equations 3.10 and 3.11). The 2D method contains progressive one dimensional decompositions; one operating on image rows and another on image columns as:

$$\phi(x, y) = \phi(x)\phi(y) \quad (3.12)$$

where, both $\phi(x)$ and $\phi(y)$ are 1D wavelets and the combination of them can produce a 2D wavelet.

The wavelet transform of an image, as shown in Figure 3.20.a, is also a multi-resolution description as at each stage the resolution (number of pixels) is reduced in a pyramid manner. The wavelet transform uses scaled and translated versions of a prototype wavelet as basic functions to represent a signal. The three high-pass filtered datasets (high-low, low-high, and high-high) represent the wavelet transforms details components at that level of scale of the transform. The low-pass filtered dataset (low-low) is the approximation components at that level of scale and the four sets of components have four times fewer elements than the original data set. The approximation components can now be used as the sampled data input for another pair of wavelet filters, identical to the first pair, thus generating another set of details and approximation components at the next lower level of scale and the proposed method requires three and four steps of decomposition for a missing macro block size of 8×8 and 16×16 , respectively.

The decomposition of an image signal can be viewed as low-pass-horizontal and low-pass-vertical filtering, high-pass-horizontal and high-pass-vertical filtering, the low-pass-horizontal and high-pass-vertical filtering and the high-pass-horizontal, low-pass-vertical filtering, respectively (Swati, Malviya & Lade, 2013) (see Figure 3.20). During this transformation each stage of the decomposition can be expressed as a product of the input image and a filter matrix:

$$cA_{j+1} = [LL_{j+1}]cA_j \quad (3.13)$$

$$cH_{j+1} = [LH_{j+1}]cA_j$$

$$cV_{j+1} = [HL_{j+1}]cA_j$$

$$cD_{j+1} = [HH_{j+1}]cA_j$$

where , is the decomposition stage and initially, $cA_0 = image$.

3.5.2 Components of Wavelet Decomposition: Approximation and Details

An example of image decomposition is shown in the top left of Figure 3.20 and is decomposed into four quadrants with different interpretations (HH, HL, LH and LL).

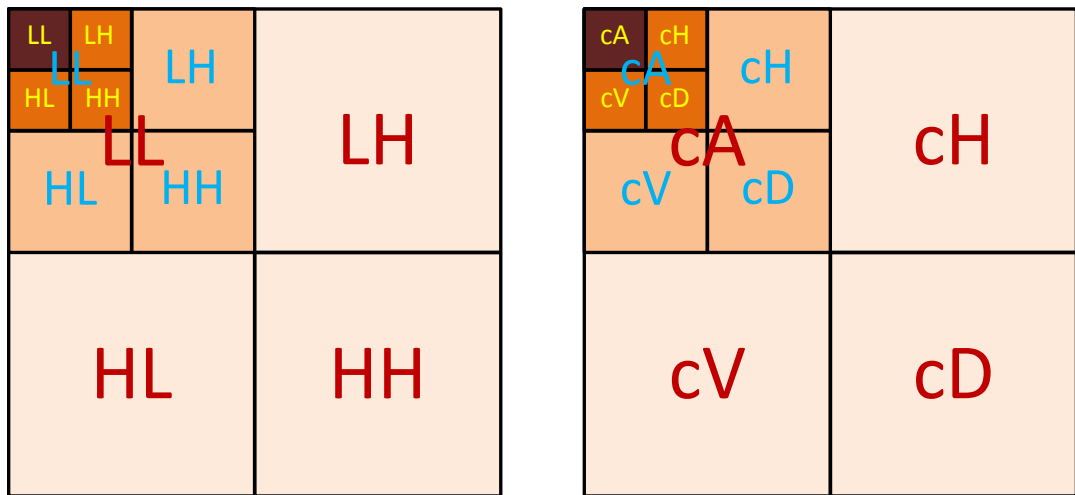
The successive application of two-dimensional DWT leads to a decomposition of the approximation coefficients at level j in four components: the approximation at level $j + 1$, and the details in three orientations (horizontal, vertical, and diagonal) that are related mostly to image edges. It can be seen in Figure 3.20, top right, that the two-dimensional wavelet decomposition computes the approximation coefficients matrix cA and details coefficients matrices cH , cV , and cD (horizontal, vertical, and diagonal, respectively), obtained by wavelet decomposition of the input image.

LL (cA) : are all the filtered coefficients (with half the resolution) obtained from applying a low-pass filter $h[n]$ along rows and then columns, gathered on the upper left section.

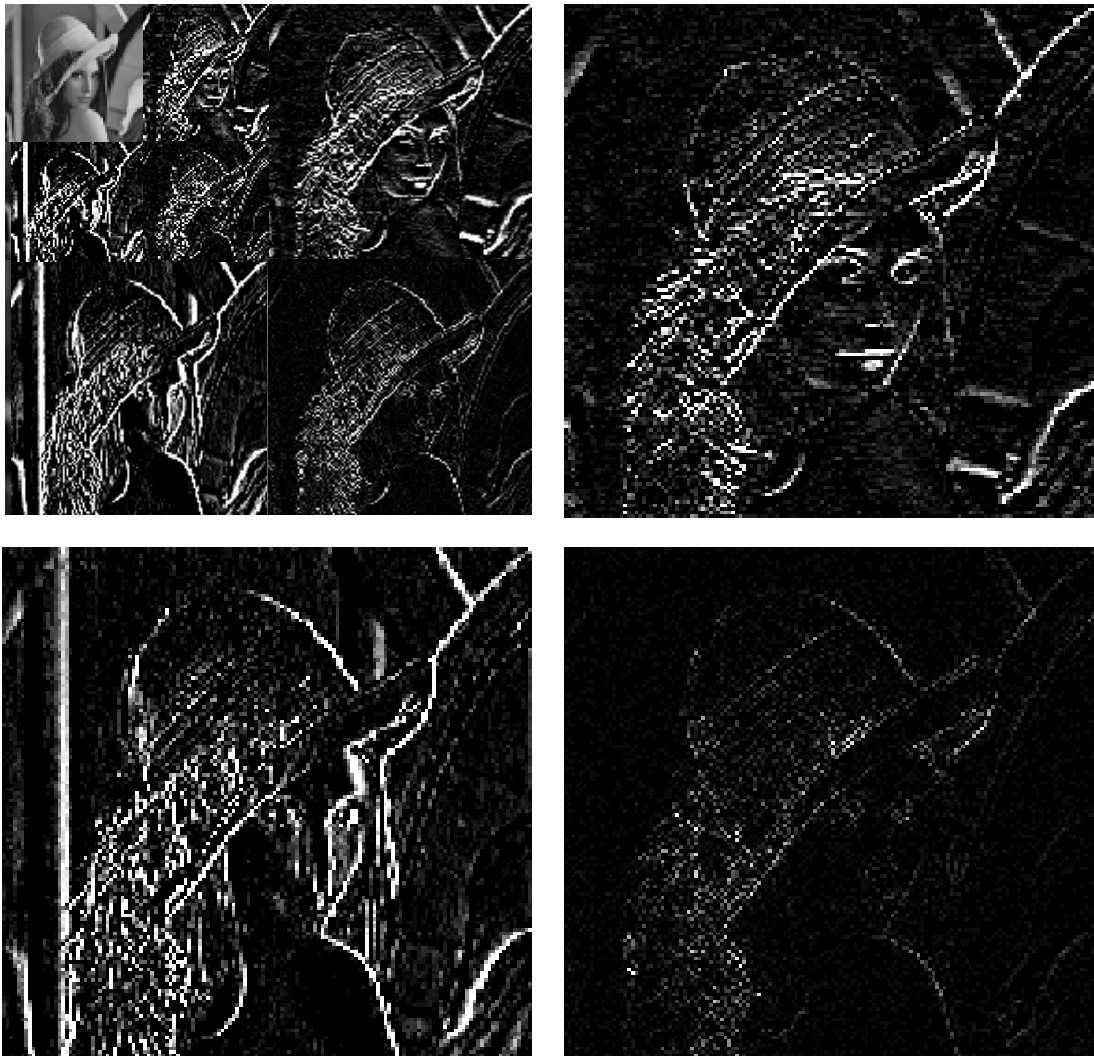
LH (cH) and HL (cV) represent the energy of the wavelet coefficients in the horizontal and vertical direction, respectively: filtered coefficients obtained from applying $h[n]$ and $g[n]$ along the rows and columns of the image. The LH section includes horizontal edges, and the HL section shows vertical ones.

HH (cD) represent the energy of the wavelet coefficients in the diagonal direction: all filtered coefficients obtained from applying a high-pass filter $g[n]$, along rows and then columns. HH coefficients illustrate the edges in the diagonal directions.

The wavelet transform is a multi-scale process and Figure 3.21 shows the three-stage wavelet pyramid image decomposition on the Lena image. In each level of decomposition, cA , cH , cV and cD are illustrated, which represent LL, LH, HL and HH respectively. Moreover, Figure 3.21 demonstrates the three-stage wavelet pyramid image decomposition and its application to the Lena image on four down-sampling levels from 512×512 to 64×64 .



(a)



(b)

Figure 3.20: (a) Wavelet pyramid image decomposition coefficients in three- stages; and (b) its application to Lena image of size 512×512.

The wavelet transform might be more suitable for transient (fast changing) audio/image/video data compression. For, information data may store in less space than the other alternative methods such as DCT. However it is not effective with smooth data, for which the traditional techniques like the Fourier transform are more beneficial.



Figure 3.21: Block diagram of the three-stage wavelet pyramid image decomposition and its application to the Lena image.

3.6 Edge Detection

Edge detectors are one of the main tools employed in digital image processing and have been applied in several practical applications including in this thesis. The main aim of edge processing applications is to reduce the amount of data, while keeping the structural information of the image, in order to use it in the next image processing procedure. One of the first applications is attributed to Marr and Hildreth (1980) and is based on the zero-crossing of the Laplacian of Gaussian in each image. The main aim of using edge detectors is defining different segments, regions and objects within an image by finding the presence and location of the major edges.

Theoretically, an edge is a step discontinuity that in its classical form is non-differentiable $\left. \frac{dy}{dx} \right|_{edge} \rightarrow \infty$. However, in practice edges are sharp transitions that occur over a non-zero time-space zone and hence, are differentiable. Edges tend to

define the boundary between different signal events or objects (such as the alphabet in this thesis). Edge detection is used to reduce an image to a skeleton copy composed of the boundaries of the objects within the image.

An edge can be described as a sharp change in the intensity levels of an image (Basu, 2002). In general, edges in an image are a group of joint pixels, which define the boundaries between image regions, such as between various objects and the background. Hence, an edge detection process separates the image into a region of discontinuity by using image segmentation (Canny, 1986).

Edge detection methods simplify image processing by eliminating the amount of unnecessary data, and provide vital structural information regarding boundaries within the image (Canny, 1986). In general, there are various types of edge detectors, which use alternative differential operators. Laplacian based techniques and gradient based methods are two widely used techniques, with Marr-Hildreth being one of the former and those of Sobel, Prewitt, Robert and Canny examples of the latter.

Laplacian edge detectors: were first proposed by Marr and Hildreth (Asghari Oskoei & Hu, 2010). They are able to locate the sharp and sudden changes in intensity of the image, consequently edges can be found by this method. Laplacian edge detection techniques are based on finding the zero crossing points in the second derivations of the image. The Laplacian $L(x, y)$ of an image is given by (Seerha & Kaur, 2013):

$$L(x, y) = \frac{\partial^2 x}{\partial x^2} + \frac{\partial^2 y}{\partial y^2} \quad (3.21)$$

where, x and y are the pixel intensity values of the pixel $A(x, y)$.

Gradient (Gaussian) based edge detector techniques: are based on finding the maximum and minimum values for the first derivative of the image along the x and y directions (Basu, 2002), with these derivatives being a measure of the change in pixel intensity with distance in the x and y directions, as can be seen in the Equation 3.14. Then, the gradient values at each pixel in the image need to be calculated, in order to obtain the gradient of the image as a whole.

$$G_x = \frac{\partial f(x, y)}{\partial x} = \frac{f(x + d_x, y) - f(x, y)}{d_x} \quad (3.14)$$

$$G_y = \frac{\partial f(x, y)}{\partial y} = \frac{f(x, y + d_y) - f(x, y)}{d_y}$$

where, d_x and d_y represent a small change in distance in the x and y directions. As d_x and d_y are the number of pixels between a pair of pixels in a discrete image, then those values can be set to one (for neighbouring pixels) and as a result, Equation 3.15 is changed to (Seerha & Kaur, 2013):

$$d_x = 1 \quad \rightarrow \quad G_x = f(x + 1, y) - f(x, y) \quad (3.15)$$

$$d_y = 1 \quad \rightarrow \quad G_y = f(x, y + 1) - f(x, y)$$

There are two main functions in the gradient-based edge detection methods. The first is the gradient magnitude (Equation 3.16), which calculates the change in the gradient value at the point (x, y) .

$$G(x, y) = \sqrt{(G_x^2 + G_y^2)} \quad (3.16)$$

Second, the gradient orientation or gradient direction is calculated by Equation 3.17 (Petrou & Petrou, 2010):

$$\theta(x, y) = \tan\left(\frac{G_y}{G_x}\right) \quad (3.17)$$

where, $\theta(x, y)$ represents the direction of the gradient. Note, the edge direction is orthogonal to the gradient vector when find a pixel's gradient (Petrou & Petrou, 2010).

Each of the Equations 3.15 describes a one dimensional method for calculating the gradient of change along the x and y directions using a one dimensional mask vector $[1, -1]$. The calculations of the differential operators along these directions can be extended to a two dimensional matrix (mask of size 2×2) for including the diagonal edges. The first 2D filter (as can be seen in Equation 3.18) was proposed by Roberts

in 1965 (Rashmi, Kumar & Saxena, 2013). As an improvement, a mask of size 2×2 with more information about the directions has been used and defined as:

$$G_x = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \quad (3.18)$$

$$G_y = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

Even though those 2D edge detection methods are simple to implement and not time consuming, they are more sensitive to noise and not symmetric to the center. To address this, a variety of modern edge detection methods developed since the 1960s use two-dimensional 3×3 masks of the form:

$$G_x = \begin{bmatrix} -1 & -k & -1 \\ 0 & 0 & 0 \\ 1 & k & 1 \end{bmatrix} \quad (3.19)$$

$$G_y = \begin{bmatrix} -1 & 0 & 1 \\ -k & 0 & k \\ -1 & 0 & 1 \end{bmatrix}$$

The advantage this mask offers is smoothing in one direction and taking the difference in another direction. The parameter K can have different values and then the filter will be changed to different edge detectors.

Selecting $k = 1$, represents a mask of a Prewitt edge detector which was put forward in 1970 by the author as (Asghari Oskoei & Hu, 2010):

$$G_x = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad (3.20)$$

$$G_y = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

The Sobel method has a similar mask with a value of $k = 2$, which provides for better coping with noise, but it increases the computational complexity.

The first group of edge detection methods set out above only utilises the local gradient operation and is able to find edges with some specific orientation. Hence these methods are not capable of detecting edges in a noisy image and so some do

not provide high performance in the case of having such an image. Under these circumstances, they usually increase the probability of producing a false edge and are incapable of the diagnosis of sharp edges.

As system performance will be hampered by edge detector errors, finding the best operator is vital in order to achieve the best result. Figure 3.22 shows the performance results on the Lena image after passing it through different edge detection methods. The outcomes show the best result belongs to the Canny edge detector, whilst Sobel and Prewitt are approximately the same and the worst result comes from the Roberts edge detection. Of all the proposed edge detection methods, up until now, the Canny nearly always provides the best results and hence, it is widely

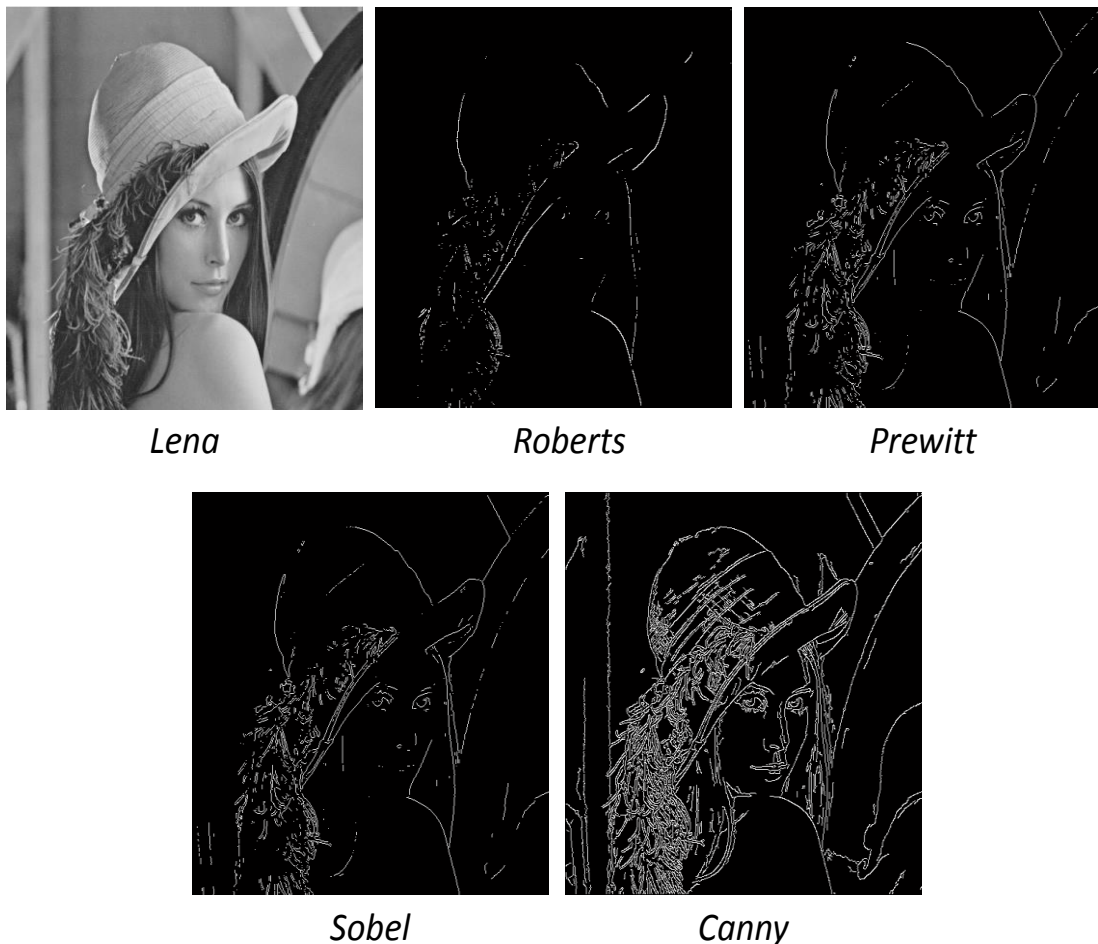


Figure 3.22: Experiments using different edge detection methods on Lena (512×512); note Canny has two threshold parameters in comparison to one threshold parameter for other methods.

employed (Canny, 1986; Asghari Oskoei & Hu, 2010; Seerha & Kaur, 2013; Rashmi, Kumar & Saxena, 2013). In addition, Canny uses Sobel as a computational filter for finding horizontal, vertical and diagonal edges (Seerha & Kaur, 2013). Hence, in the following, two prominent edge detectors, Sobel and Canny, are introduced.

3.6.1 Sobel Edge Detection

Sobel is a well-known classic edge operator proposed by Irwin Sobel in 1968 (Asghari Oskoei & Hu, 2010), which uses a mask matrix as a 2D spatial gradient convolution in order to find the edges. A Sobel filter applies its convolution mask in two directions: vertical and horizontal in the image.

The general mask was proposed as Equation 3.19 and if the parameter K is replaced with 2 then we have the 2D Sobel mask for vertical and horizontal direction (Equation 3.22) (Petrou & Petrou, 2010).

$$S_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \quad S_y = S_x^T = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (3.22)$$

Filtering of the source image A with the differential operators, S_x and S_y , yields two differentially-enhanced images G_x and G_y (Equation 3.23) and the vector containing these two values shows the direction of the greatest rate intensity change at (x, y) .

$$G_x = S_x * A \quad \text{and} \quad G_y = S_y * A \quad (3.23)$$

where, the operator $*$ denotes the 2D convolution or filtering operation. The two differentially-enhanced images, G_x and G_y , can be combined to give the gradient magnitude operator, defined as:

$$G = \sqrt{G_x^2 + G_y^2} \quad (3.24)$$

Figure 3.23 and 3.24 illustrates all G , G_x and G_y , which are images with the same size as the original.

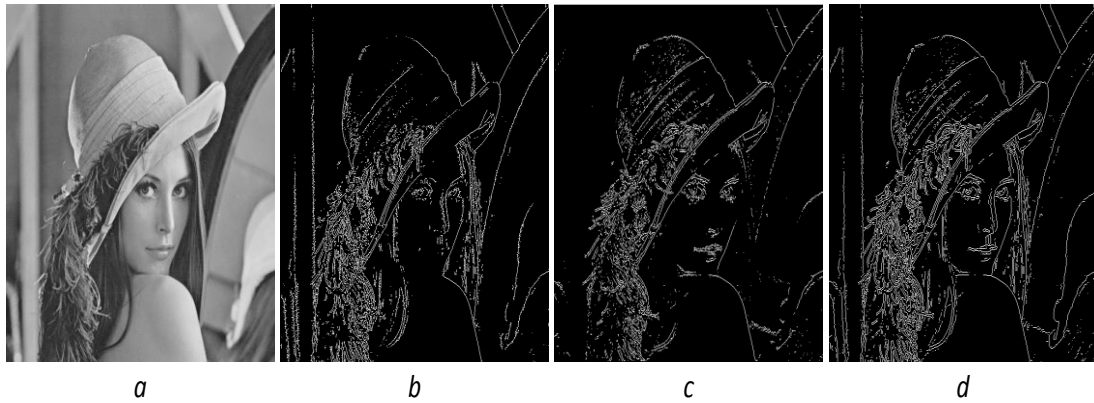


Figure 3.23: Application of the Sobel filter: a) original Lena (512×512), b) x direction (G_x), c) y direction (G_y), and d) the combined gradient direction.

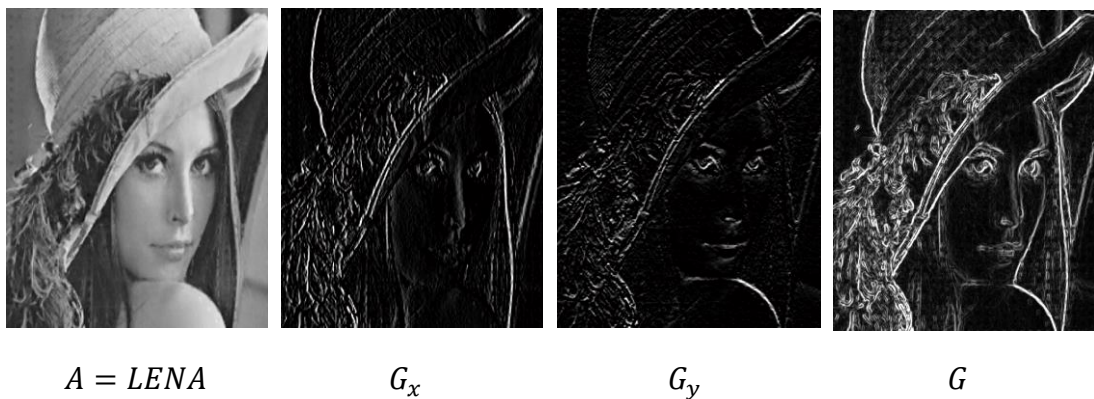


Figure 3.24: Application of the Sobel filter to zoomed Lena (512×512) in x , y and the combined gradient directions.

3.6.2 Canny Edge Detection

The Canny edge detection technique was proposed in (1986) by Canny. The purpose of his work was to present a better edge detection method able to provide the optimal detection of edges with less false detection, whilst preserving sharp edges. Nowadays, Canny is one of the main edge detection methods employed (Seerha & Kaur, 2013).

Canny uses a multi-stage algorithm employing pruning, linking and thinning. There are three main desired criteria involved in the Canny edge detection operator: good detection (detecting the correct edge and not having a false detection where there are none), good localisation (minimum distance from the detected edge and the actual edge) and having one representative for each edge (in some cases there is more than one detected edge) (Gonzales & Woods, 2008).

Canny utilises an exponential function and then, applies the first derivative of the Gaussian function to achieve high performance. The Gaussian filter operates as a smoothing function and subsequently, the first derivative is applied to the outcomes.

$$f_{\sigma}(x) = \frac{dG_{\sigma}(x)}{dx} = -k \frac{x}{\sigma^2} \exp\left(-\frac{x^2}{\sigma^2}\right) \quad (3.25)$$

where, $G_{\sigma}(x)$ and $f_{\sigma}(x)$ denote the 1D Gaussian function and its derivative, respectively.

For a 2D edge, Canny uses the feature of a Gaussian filter and utilises two separate 1D filters in the horizontal and vertical directions. By using two 1D filters the complexity might be reduced (Asghari Oskoei & Hu, 2010).

$$f_{\sigma}(x, y) = [f_{\sigma}(x) * G_{\sigma}(y) \quad G_{\sigma}(x) * f_{\sigma}(y)] \quad (3.26)$$

where, $f_{\sigma}(x, y)$ denotes the 2D optimal filter and $G_{\sigma}()$ and $f_{\sigma}()$ denote the 1D Gaussian function and its derivative, respectively.

Signal noise ratio (SNR) is utilised in order to measure the detection ability, which has a reverse relation with the probability of detecting the false edge in the image. The higher the SNR value implies the less the false detection probability and this value can be calculated as:

$$SNR = \frac{\left| \int_{-W}^W G(-x) f(x) dx \right|}{n_0 \sqrt{\int_{-W}^W f^2(x) dx}} \quad (3.27)$$

where, $G(x)$, $f(x)$ and n_0 represent the edge, optimal operator and root mean square (RMS) of the noise, respectively.

The localisation equation is derived from Equation 3.28 and the maximisation of the following equation (based on the derivatives of the edge and the operator) provides the optimal localisation for the proposed method (Basu, 2002).

$$Localization = \frac{\left| \int_{-W}^W G'(-x) f'(x) dx \right|}{n_0 \sqrt{\int_{-W}^W f'^2(x) dx}} \quad (3.28)$$

And finally, different edge detection might be provided for a single edge and it is necessary to eliminate the false ones.

The Canny proposed method demonstrated that for a 1D signal, the use of the Gaussian filter with variance σ is sufficient to obtain improved results. Similarly, with a 2D signal, two filters are utilised separately for horizontal and vertical direction.

In addition, to reduce spurious edge detection, adaptive thresholding with two different thresholds are used.

The implementation of the Canny operator can be expressed in the following stages (Gonzales & Woods, 2008) (Figure 3.25):

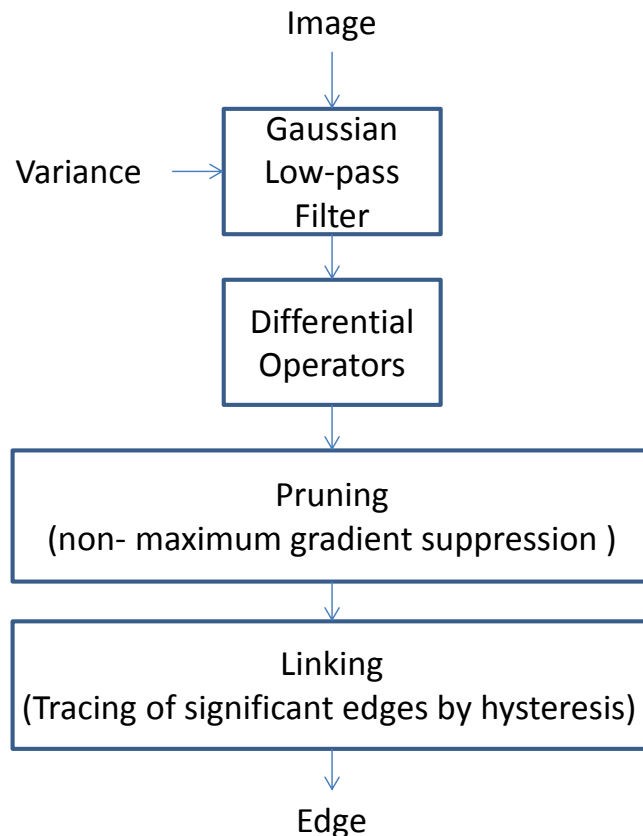


Figure 3.25: Diagram of the Canny edge detector.

- 1) A Gaussian filter used as a pre-processor to smooth out random fluctuations and noise that may result in the detection of spurious edges (because some

images might include noise and it can decrease the ability of detection). The variance of the filter can be empirically changed to obtain the best value;

- 2) Computation of horizontal, vertical and diagonal (finding gradients) edges using a difference filter, such as the Sobel filter and calculating gradients in both the x and y directions (by finding the maximum intensity changes within the image);
- 3) Suppression of insignificant non-maximum gradient edges by using the local gradient;
- 4) Tracing of significant edges through image and hysteresis (both high and low threshold) thresholding. After suppression of non-maximum gradient edges, each pixel is marked by their values. Some are correct edges and the rest might be detected because of noise or location in the image. Therefore, thresholding can be used to eliminate the ones which have not reached the value of the threshold. If the gradient magnitude of a pixel is above the high threshold then it is marked as an edge. If this magnitude is above the low threshold and also is next to an edge, it is selected as an edge, but if this is not the case, it is not selected. Finally, if the gradient magnitude of a pixel is less than the low threshold it is not counted as an edge.

Chapter 4

Image Gap Restoration Using a Multi-scale DCT Pyramid

Chapter 4

4. Image Gap Restoration Using a Multi-scale DCT Pyramid

4.1 Introduction

Multi-scale methods are one of the major image processing approaches employed to obtain efficient estimation and coding solutions. Often they combine the division of signals into progressively smaller frequency bands and time/space scales.

In this process a signal is divided into a number of sub-signals by multi-scale techniques in order to analyse the different representation levels, and as a result these methods are able to obtain the beneficial features among all the provided information. In addition, the transformation creates an information pyramid, where successive down-sampled layers of the signal have less detail (low-pass processes), which leads to easier interpolation and estimation, hence the pyramid can be used to facilitate the estimation strategy and reduce the computational complexity (Dorini & Leite, 2009).

Sub-band transform coding (e.g. filters, DCT, FFT) is the main component of the successive layers of a pyramid, being a type of transform coding that divides a signal into a number of different frequency bands and applies the procedure on each one independently. In tree-structured sub-band methods, such as quadrature mirror filters (a filter whose magnitude response is a mirror image about $\frac{\pi}{2}$ of that of another filter), a signal is progressively split into sub-bands and down-sampled and then further split into narrower bands, which produces a family of multi-scale signals. The general form of a multi-scale pyramid decomposition is illustrated in the block diagram of Figure 4.1. A 2D transform or a set of 2D filters decomposes the signal into low and high frequency regions. The low frequency part of the image is down-

sampled and further decomposed into sub-band frequency regions. The process is repeated several times depending on the application and the size of the image blocks.

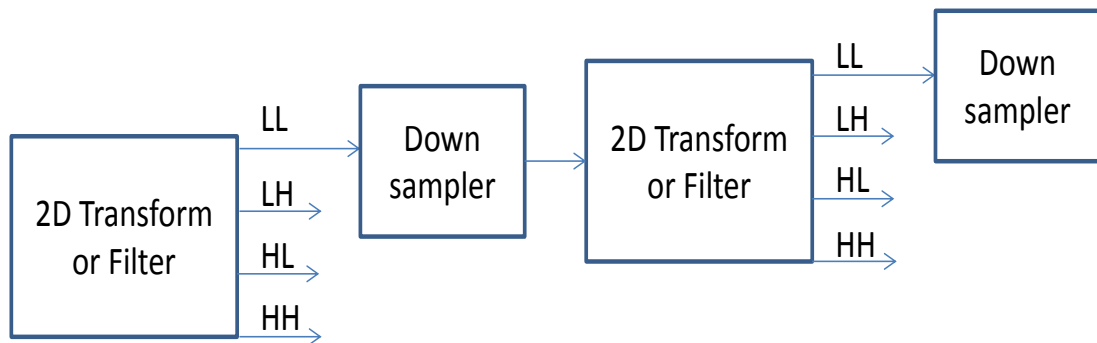


Figure 4.1: Multi-scale pyramid decomposition.

Wavelets and scale-space filtering are the other two techniques that use the multi-scale approach. Wavelet is a powerful mathematical time-frequency-transformation technique in signal analysis, which can capture both frequency and location in time, this being its advantage over a Fourier transform (Mohlenkamp, n.d.). The Scale-space filtering technique is a framework for multi-scale signal representation for handling image structures at different scales using Gaussian masks over a sequence of sizes, which are subsequently transformed into a tree frame that can provide a complete qualitative framework covering all scales of the observation (Witkin,1984).

The multi-scale discrete cosine (DCT) transform pyramid and discrete wavelet (DWT) transform are used in this project and are explained in the following sections.

4.2 Multi-Scale Discrete Cosine Transform Pyramid

In the multi-scale DCT method, the transformation of the image blocks into the constituent sub-band regions is achieved using the DCT transform. The multi-scale DCT pyramid processing method illustrated in Figure 4.2 and Figure 4.3 (four and five layers of pyramid, respectively) progressively decomposes image macro blocks (MBs) into four spectral quadrants, LL, LH, HL, HH, where L and H denote the low and high frequency halves of the spectrum, respectively.

At the first stage of DCT decomposition each macro block is transformed into the DCT domain, the low frequency LL quadrant is selected, which is then transformed back to the image domain to produce an image, the bandwidth and scale of which are a quarter of that of the input image. At each subsequent down-sampling and decomposition stage, the LL quadrant is further decomposed into four spectral quadrants, the new down-sampled LL is selected and the inverse is transformed back to the image until the macro block is reduced to a single pixel. For a macro block of size 8×8 , three stages of decomposition and down-sampling reduces the macro block to one pixel, as shown in Figure 4.2. Whereas for a macro block of size 16×16 , four stages of decomposition and down-sampling reduce the macro block to one pixel as illustrated in Figure 4.3. The name pyramid arises from the fact that reduced-scale image layers can be thought of as forming a structure of this shape.

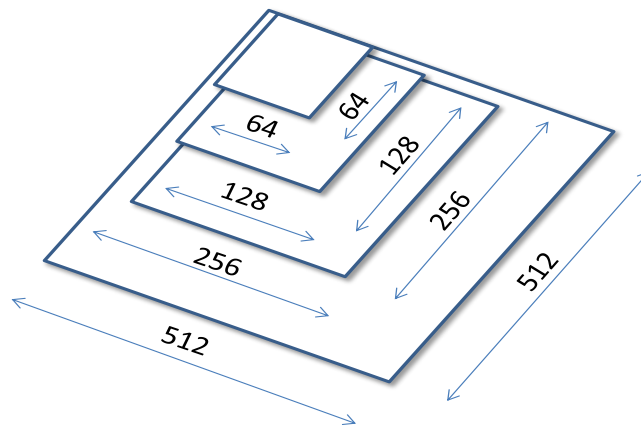


Figure 4.2: Four layer pyramid decomposition using 8×8 macro blocks.

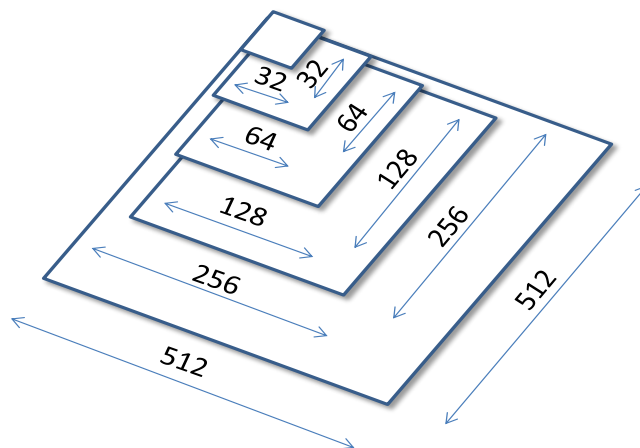


Figure 4.3: Five layer pyramid decomposition using 16×16 macro blocks.

This multi-scale DCT decomposition method is similar to wavelet filter structures, however, the DCT functions are used here as the basis function.

The multi-scale 2D-DCT of an $M_s \times N_s$ matrix, A , at the base layer is defined as:

$$B_{pq0} = a_p a_q \sum_{m=0}^{M_s-1} \sum_{n=0}^{N_s-1} A_{mn} \cos \frac{\pi(2m-1)p}{2M_s} \cos \frac{\pi(2n-1)q}{2N_s} \quad (4.1)$$

where, the subscript s denotes the pyramid layer for a block size of $M \times N$, for $s = 0, 1, 2, 3$ block size $M \times N$ are $M_3 = N_3 = 8$, $M_2 = N_2 = 4$, $M_1 = N_1 = 2$, $M_0 = N_0 = 1$. $0 \leq p \leq M_s - 1$, $0 \leq q \leq N_s - 1$ and:

$$a_p = \begin{cases} 1/\sqrt{M_s}, & p = 0 \\ \sqrt{2/M_s}, & p \neq 0 \end{cases} \quad \text{and} \quad a_q = \begin{cases} 1/\sqrt{N_s}, & q = 0 \\ \sqrt{2/N_s}, & q \neq 0 \end{cases}$$

Note that, as shown in Figure 4.4 and Figure 4.5, the down-sampling by a factor of a half (2:1) is performed by simply retaining a quarter of the low-frequency index coefficients, the LL quadrant, and discarding the remaining three quarters, higher index, coefficients.

Note also that a single DCT is sufficient to produce a set of layered pyramid coefficients composed of the subsets of the DCT coefficients. However, separate DCT and IDCT might be required at each sub-processing stage and in the recombination stages, as shown in Figure 4.4 and Figure 4.5.

The 2D-DCT coefficients of the layers 1 to 3 for block of size 8×8 are defined in terms of the base layer DCT coefficients as:

Pyramid layer 1 coefficients for a block of size 4×4 ($p = 0:3, q = 0:3$) are extracted from the DCT of the base layer 0:

$$B_{pq1} = B_{pq0}$$

Pyramid layer 2 coefficients for a block of size 2×2 ($p = 0:1, q = 0:1$) are extracted from the layer 1 and layer 0 DCT:

$$B_{pq2} = B_{pq1} = B_{pq0}$$

Pyramid layer 3, the apex coefficients of size 1×1 ($p = 0, q = 0$) are extracted from the layer 1 DCT or equally from layer 2 and layer 0:

$$B_{pq3} = B_{pq2} = B_{pq1} = B_{pq0}$$

During the re-composition stages, starting from the apex of the multi-scale pyramid, image up-sampling by a factor of two (1:2) is performed by a combination of a process of zero-padding of the 2D-DCT coefficients and the subsequent application of inverse 2D-DCT.

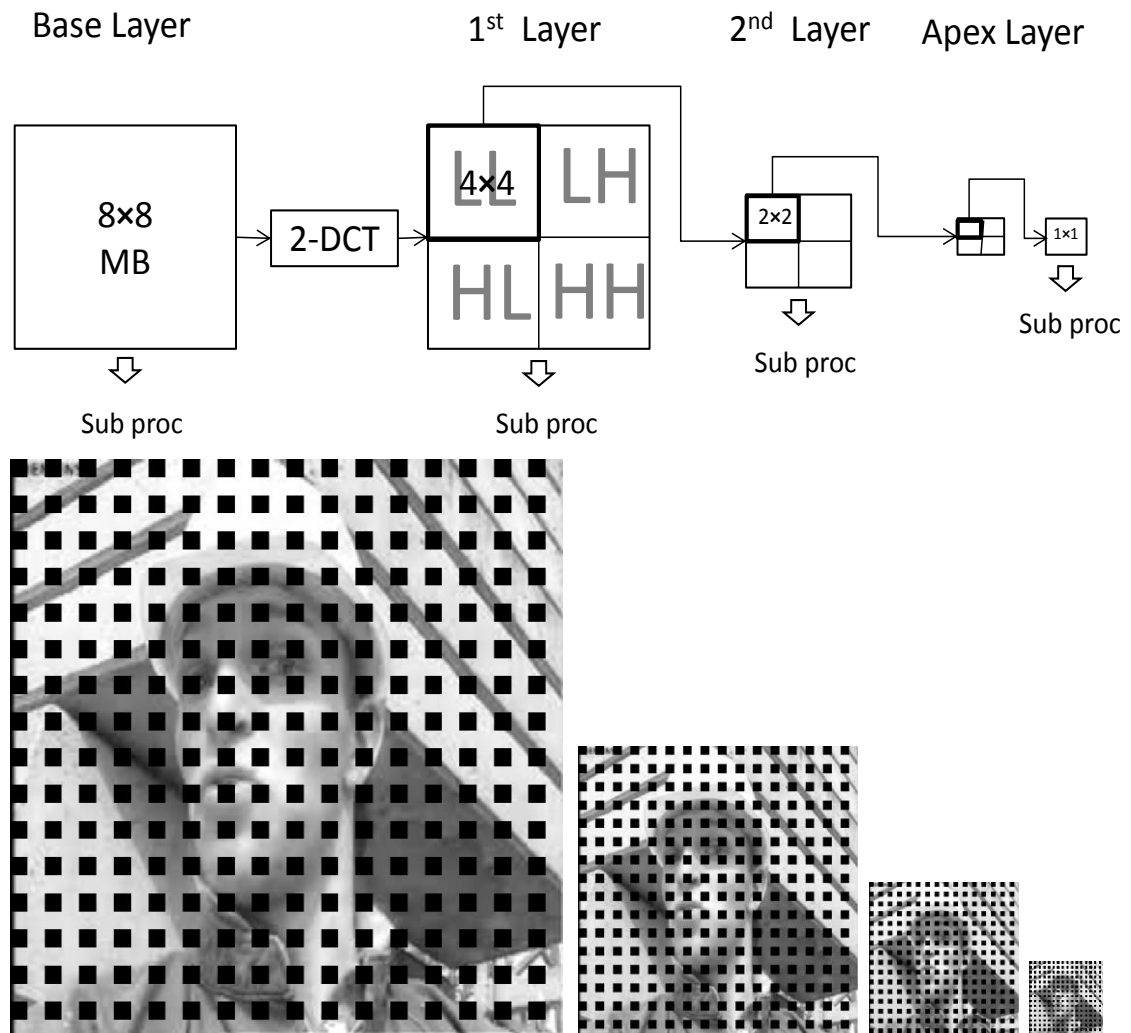


Figure 4.4: Block diagram of the three-stage DCT pyramid image decomposition and its application to the Foreman image for a missing block size of 8×8 .

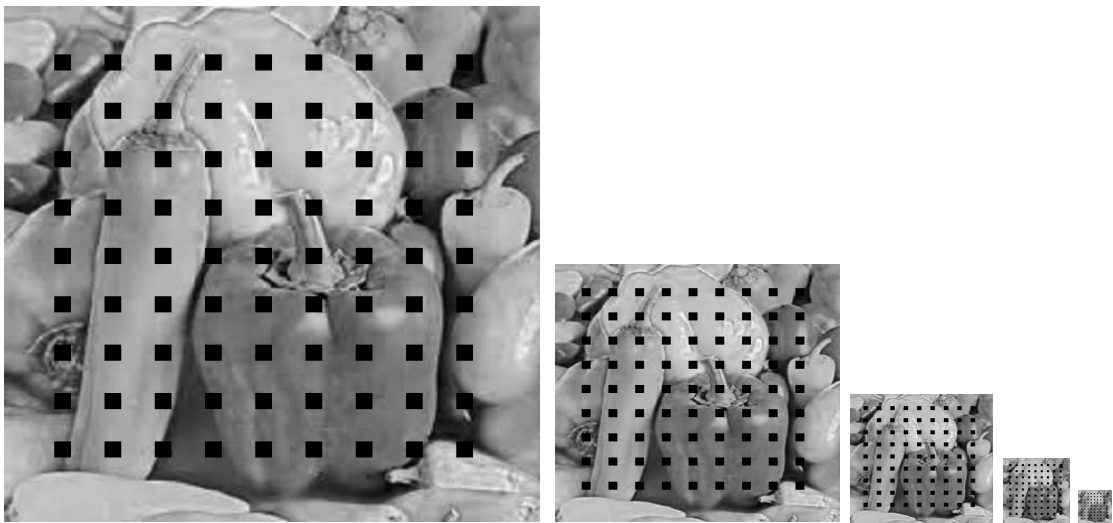
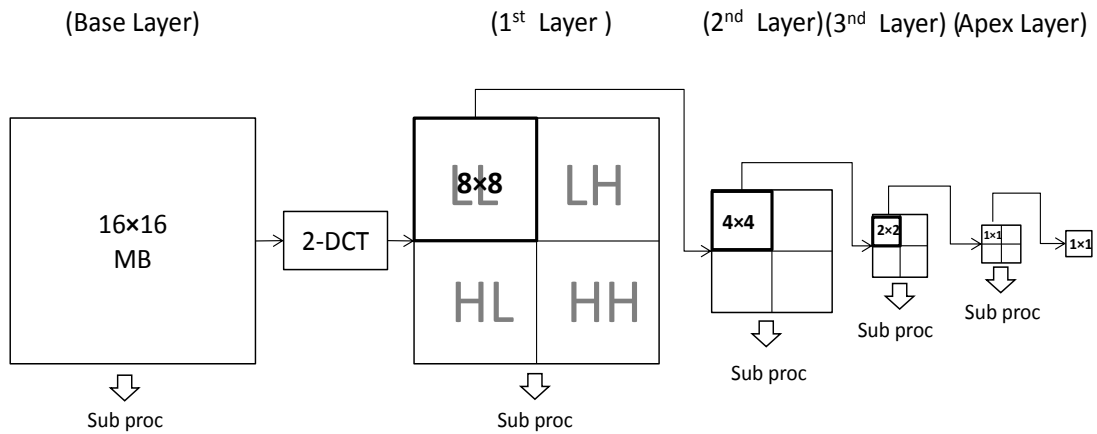


Figure 4.5: Block diagram of the four-stage DCT pyramid image decomposition and its application to the Peppers image for a missing block size of 16×16 .

As illustrated in Figure 4.5, four stages of a DCT pyramid are required for image decomposition with a block size of 16×16 . Then, after applying down-sampling and required image processing it is vital to revert the image back to the original size. Figure 4.6 shows the visual impact of reconstruction to the size of 512×512 from down-sampled signals, where this is performed directly from each layer of the pyramid.

There are different types of reconstruction of down-sampled image to the original image size. Hence, the performance of different interpolation methods (nearest neighbour (NN), linear, cubic and spline interpolation) for MB sizes 8×8 and 16×16 is evaluated (to reconstruct the image to the image original size from the apex layer by using the interpolation methods only), in order to compare the result and select the best method for this research (Table 4.I and 4.II).

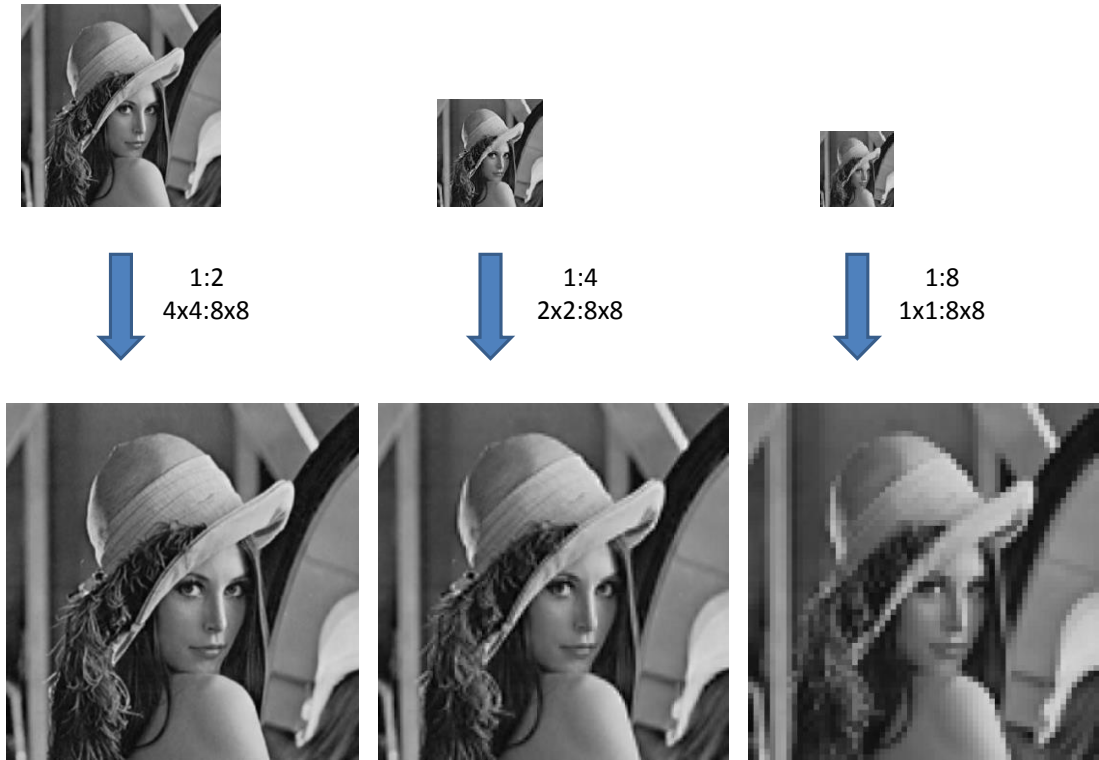


Figure 4.6: Reconstruction from individual pyramid layers: layer 1 (down-sampled by 2×2), layer 2 (down-sampled by 4×4) and layer 3 (down-sampled by 8×8).

Table 4.I: Performance comparisons of different interpolation methods (NN, linear, cubic and spline interpolation) for MB size = 8×8 , on Lena original image size 512×512 .

Down sample factor	Image size	Performance measure	Interpolation methods			
			<i>NN</i>	<i>Linear</i>	<i>Cubic</i>	<i>Spline</i>
2	256×256	PSNR (dB)	33.63	38.07	41.16	42.42
		SSIM	0.9525	0.9012	0.9441	0.9894
4	128×128	PSNR (dB)	27.74	30.08	31.44	31.99
		SSIM	0.8281	0.9070	0.9490	0.9134
8	64×64	PSNR (dB)	23.77	24.91	25.74	25.99
		SSIM	0.6703	0.9012	0.9441	0.7752

Table 4.I shows the PSNR and SSIM for four different interpolation methods, for cases where images were interpolated to the original size of 512 having been down-sampled by factors of 2 (256×256), 4 (128×128) and 8 (64×64), with a macro block of size of (8×8). As the down-sampling rate and hence information

loss increases, the gap in the performance of the linear interpolator (the worst) and the spline interpolator (the best) decreases from 8.79 dB to 2.22 dB in the macro block of size of (8×8) .

Table 4.II: Performance comparisons of different interpolation methods (NN, linear, cubic and spline interpolation) for MB size = 16×16 , on Lena original image size 512×512 .

Down sample factor	Image size	Performance measure	Interpolation methods			
			<i>NN</i>	<i>Linear</i>	<i>Cubic</i>	<i>Spline</i>
2	256×256	PSNR (dB)	33.44	38.02	41.20	42.61
		SSIM	0.9508	0.9679	0.9744	0.9896
4	128×128	PSNR (dB)	27.42	30.12	31.42	32.00
		SSIM	0.8164	0.8560	0.8789	0.9100
8	64×64	PSNR (dB)	23.53	24.94	25.69	25.93
		SSIM	0.6547	0.6925	0.7319	0.7672
16	32×32	PSNR (dB)	20.45	21.08	21.65	21.51
		SSIM	0.5741	0.6055	0.6473	0.6606

Table 4.II demonstrates the same procedure for a macro block of size of (16×16) , and images were interpolated to the original size of 512 having been down-sampled by factors of 2 (256×256), 4 (128×128), 8 (64×64) and 16 (32×32). The performance of the linear interpolator (the worst), and the spline interpolator (the best) decreases from 9.17 dB to 1.06 dB.

4.3 Gap Restoration using Multi-scale 2D-DCT

The algorithm for implementing the proposed method for gap restoration (Figure 4.7) is as follows:

- 1) Divide the image into macro-blocks of size $M \times N$ e.g. 8×8 ;

- 2) Decompose the image macro-blocks into a DCT pyramid structure, with the apex of the pyramid representing the last stage of down-sampling, where each MB of size 8×8 or 16×16 is reduced to one pixel only;
- 3) Starting from the apex of the DCT pyramid, interpolate the decimated gap using the information from neighbouring pixels and use various methods of interpolation;
- 4) Up-sample the enhanced interpolated image, via zero-padded inverse 2D-DCT, and combine/merge with the available received samples of the same layer of up-sampling;
- 5) Go to step (1) and repeat the process for each intermediate stage of up-sampling.

The details of these sub-processes are described next in this chapter.

4.4 Alternative methods of Interpolation

Image enhancement is one of the four broad categories of functions in image processing and the other three are: image representation, image restoration/filtering and compression. Interpolation may be considered as a subset of image enhancement and it can be applied for subsequent analysis, display or in order to have a continuous space image from a discrete-space image. Hence, one of the most important steps for designing a restoration algorithm is choosing a good interpolation function in order to increase the accuracy of the estimation (Pan, 2003).

Discrete image data samples are recovered in order to have a continuous intensity surface (Su & Willis, 2004). When performing a digital image interpolation, empty spaces are being created in the source image, which are filled with the appropriate pixel values, and then the problem of approximating the intensity of the unknown pixels in an image can be solved by interpolation methods (Figure 4.8). Interpolation works by using known data to estimate values at unknown points in two dimensions and aims to achieve a best approximation of a pixel's colour and intensity based on the values at available surrounding pixels. This makes the interpolation algorithms

yield different results depending on the concept used to estimate these values (Olivier & Hanqiang, 2012).

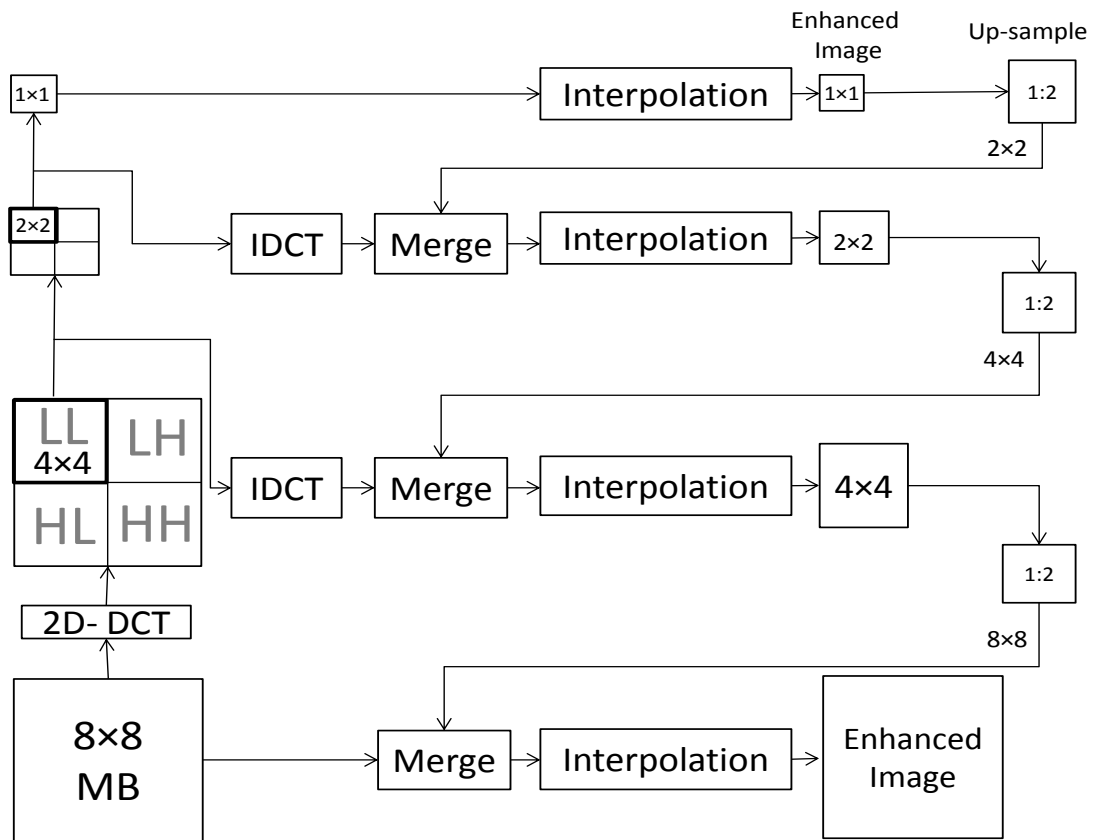


Figure 4.7: Block diagram of image gap restoration using a multi-scale DCT pyramid.

There are two types of image interpolation: adaptive and non-adaptive methods. The former involves decision making, where the filter is adapted to apply the different version of the associated algorithms on different pixels depending on the pixel type (sharp edges or smooth texture). Bayesian methods can be considered as a version of adaptive algorithms, where the mean, variance and prior information are adapted to different image segments.

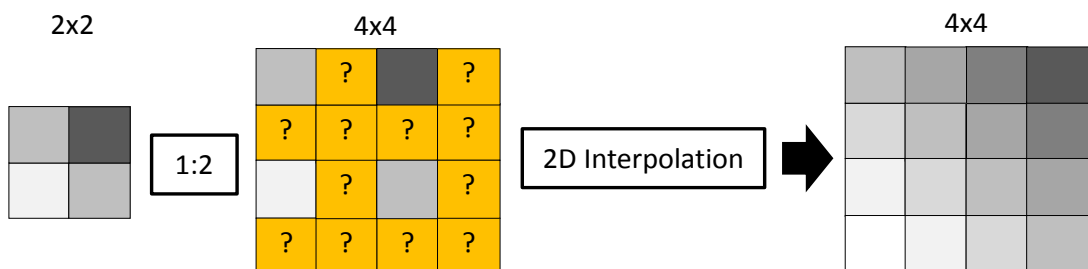


Figure 4.8: Up-sampling a 2x2 macro block into a 4x4 pixel grid by 2D interpolation (spline).

Non-adaptive techniques use the same function for all types of pixels like the most widely used methods for image interpolations: nearest, linear, cubic and spline interpolation. A further example of non-adaptive interpolation is the use of zero-padding with a Fourier or DCT transform for up-sampling (see subsection 4.4.1). These methods are completely different in terms of image resolution, speed, and theoretical assumptions.

The interpolation methods developed in this project (Chapter 5 and Chapter 6) are of the data-adaptive variety, whereby the interpolator coefficients or the regions that they operate on are adapted to the local and global edges of the segments.

In this research interpolation is used for several stages of image gap restoration:

- For estimation of the missing samples at the layers of the pyramid using local and global edges;
- For up-sampling from one pixel grid sampling layer to another higher pixel grid sampling layer (Figure 4.8);
- In the post processing stage, for producing an output that blends with the pattern of the neighbouring samples.

There are several commonly used approaches to interpolation (Pan, 2003; Su & Willis, 2004; Getreuer, 2011; Jonic & Sorzano, 2011; Olivier & Hanqiang, 2012). Each of these has its own features and yield different results. The performances of several methods are compared in this chapter in order to find the best techniques to be used in this thesis. Some of the interpolation methods are as follows:

- ❖ Low-pass filtering of the zero inserted signals, which is mostly used for up-sampling rather than estimation of missing/lost samples in a gap (see subsection 3.4.2);
- ❖ Frequency-transform, e.g. DCT, based interpolation, which can be configured and employed for both up-sampling applications and the estimation of missing gaps in a signal;
- ❖ Nearest neighbour interpolation;
- ❖ Linear interpolation;
- ❖ Cubic interpolation;
- ❖ Spline interpolation.

4.4.1 Frequency Transform, DCT Based, Interpolation

A macro block of a given size can be interpolated and up-sampled by a factor of 2 using a DCT-based interpolator, as illustrated in the block diagram of Figure 4.9. The process is composed of the following stages:

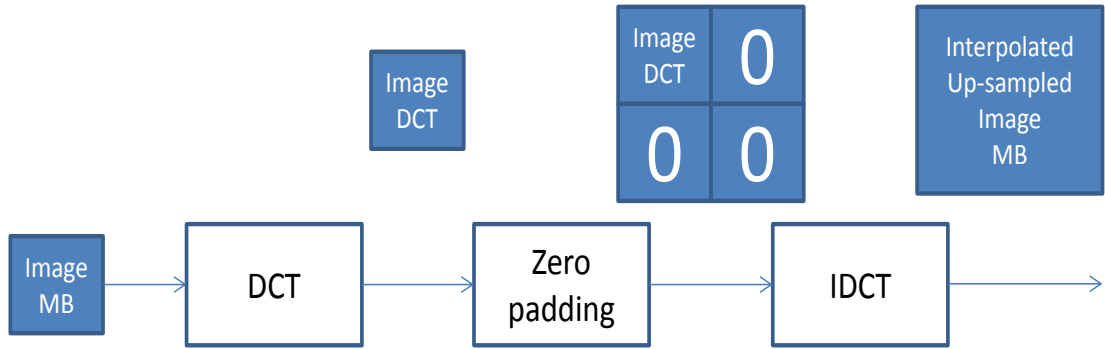


Figure 4.9: DCT based interpolation for up-sampling by factor of 2.

- 1- Apply DCT to macro blocks to obtain a 2 dimensional DCT matrix;
- 2- Zero-pad the 2D DCT matrix of each macro-block to form an up-sampled block size four times the original (assuming interpolation by a factor of 2 in xy directions);
- 3- Apply inverse DCT to the zero-padded DCT matrix to obtain an interpolated MB matrix.

In order to compute an interpolated up-sampled image macro block, the output $M_s \times N_s$ matrix is defined by Equation 4.1 and the transformation can be written in matrix form as follows:

$$Y = A_{MN} \cdot X \quad (4.2)$$

where, X refers to vectors from $M_s \times N_s$ input pixels, Y represents $M_s \times N_s$ DCT output and s denotes the pyramid layer. Each (2-D) DCT matrix (A_{MN}) can be written (by Kronecker factorisation) as two (1-D) DCT matrices (A_M and A_N matrices of sizes M and N , respectively).

$$A_{MN} = A_M \otimes A_N \quad (4.3)$$

In order to obtain an estimation for a missing block of size of $(M - 2s) \times (N - 2s)$ pixels, this has to be restored by its $(M \times N) - (2^s)$ non-corrupted border pixels, as can be seen in the Figure 4.10. Then define X_1 , the vector of $(M \times N) - (2^s)$ available pixels and X_2 , the $(M - 2s) \times (N - 2s)$ corrupted pixels and thus Equation (4.3) can be changed to:

$$Y = A_1 \cdot X_1 + A_2 \cdot X_2 \quad (4.4)$$

Note that A_1 and A_2 are matrices derived from the 2-D DCT matrix. By the same assumption, vector Y can be written as two vectors (Y_1 and Y_2) with $(M \times N) - (2^s)$ and $(M - 2s) \times (N - 2s)$ elements, respectively, and now the system of equations can be written as follows with the elements of the DCT matrix A_{MN} :

$$\begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \cdot \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \quad (4.5)$$

In order to solve the system equation, vector Y_2 is set to zero and $\det [A_{22}] \neq 0$, then the vector of the corrupted pixels X_2 is expressed as:

$$X_2 = -A_{22}^{-1} \cdot A_{21} \cdot X_1 \quad (4.6)$$

The interpolation mask matrix Z is made of $(M - 2s) \times (N - 2s)$ rows and $(M \times N) - (2^s)$ columns, which can provide estimation for the corrupted pixels from the available ones.

$$Z = -A_{22}^{-1} \cdot A_{21} \quad (4.7)$$

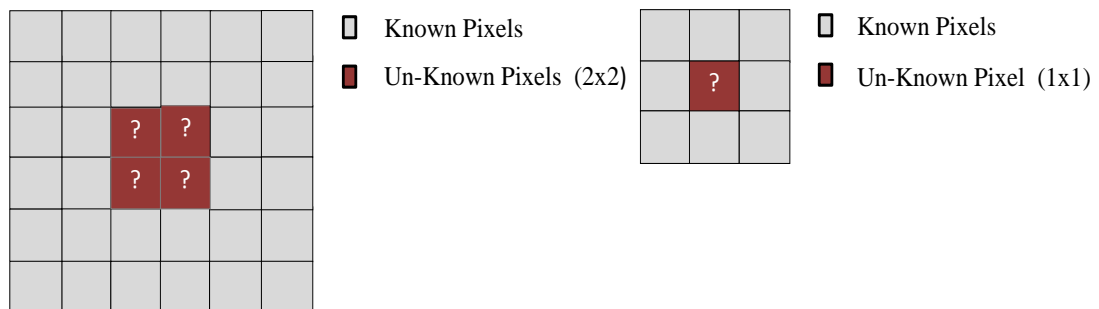


Figure 4.10: Known and missing pixels sizes (1×1) in the apex layer and (2×2) in the next layer down.

4.4.2 Nearest Neighbour Interpolation

Nearest neighbour (NN) interpolation is the simplest method of in-painting and scale magnification in terms of implementation (Pan, 2003), which gives the value to the unknown pixel from the points around the missing data in one or more dimension and the result is a piecewise-constant process. It only considers one pixel, the closest one to the interpolated missing point, as can be seen in Figure 4.11 and Figure 4.12 (Equation 4.8) and it seems to have the effect of simply making the closest pixel bigger (Figure 4.12).

$$p(x, y) = c(i, j) \quad (4.8)$$

$c(i, j)$ represents the closest point of the input samples to the unknown pixel p at position (x, y) and therefore, only one supporting point is required for the nearest neighbour interpolation. In the current work, the values of nearby known pixels are used for estimation of the output pixel values. The local one point interpolation is defined as:

$$K_1(t) = \begin{cases} 1 & \text{if } -\frac{1}{2} \leq t < \frac{1}{2}, \\ 0 & \text{otherwise,} \end{cases} \quad (4.9)$$

Figure 4. 11, shows the 4 neighbour points (i, j) , $(i, j + 1)$, $(i + 1, j)$ and $(i + 1, j + 1)$ to the unknown pixel p . The distances between (x, y) and (i, j) , $(i, j + 1)$, $(i + 1, j)$ and $(i + 1, j + 1)$ are calculated and then the values of (x, y) are set as the value of the point nearest to $p(x, y)$ (Olivier & Hanqiang, 2012).

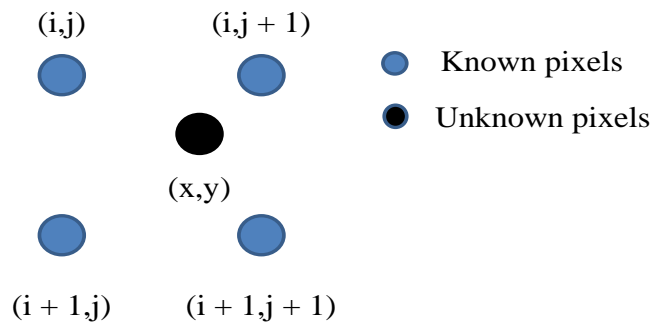


Figure 4.11: Diagram of the nearest neighbour interpolation algorithm.

Figure 4.12 demonstrates up-sampling of a 2×2 macro block into a 4×4 pixel grid by the NN interpolation. As can be seen in Figure 4.13 and Figure 4.14, the result of the nearest neighbour interpolation is not smooth in one and two dimensional interpolation and tends to increase the noise, because of the repetitions of the pixel $c(i, j)$. Consequently, strong aliasing and blurring effects are associated with the nearest neighbour method for image interpolation.

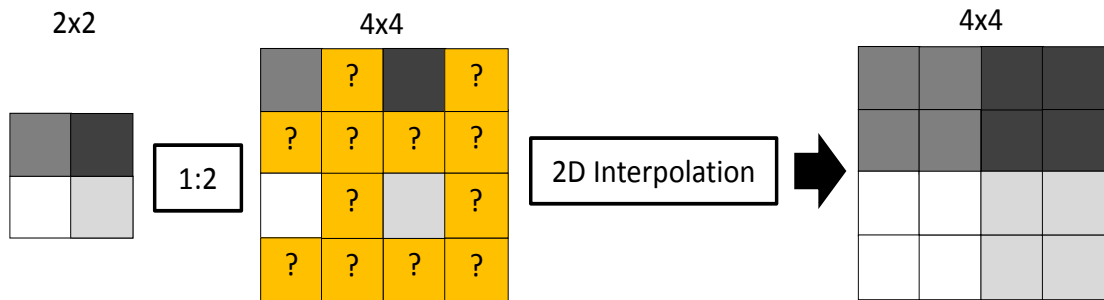


Figure 4.12: Up-sampling a 2×2 macro block into a 4×4 pixel grid by NN interpolation.

Table 4.III: Performance comparisons for an MB size= 8×8 , on Lena, Man, Peppers, Boat and Elaine for the nearest neighbour interpolation.

Interpolation Method	Performance measure	Images					
		<i>Lena</i>	<i>Man</i>	<i>Peppers</i>	<i>Boat</i>	<i>Elaine</i>	<i>Average</i>
NN	PSNR(dB)	23.77	22.09	24.81	22.30	24.02	23.39
	SSIM	0.6703	0.5178	0.6966	0.5855	0.4345	0.5809
	Time (s)	14.26	13.40	14.37	12.91	15.06	14

Table 4.III demonstrates the performance comparisons for a macro block size of 8×8 on Lena, Man, Peppers, Boat and Elaine for the nearest neighbour interpolation, where the images were interpolated to the original size of 512×512 having been down-sampled by factors of 8 (64×64).

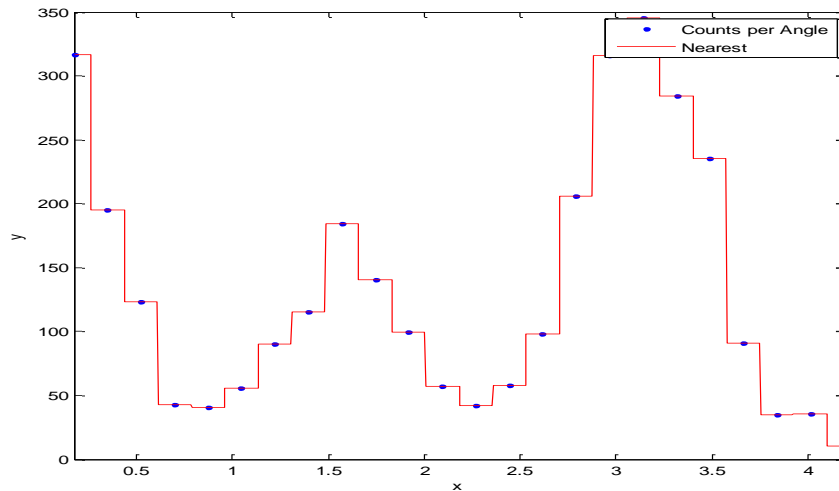


Figure 4.13: Illustration of a one-dimensional nearest neighbour (red) interpolated through a number of known data samples (blue).

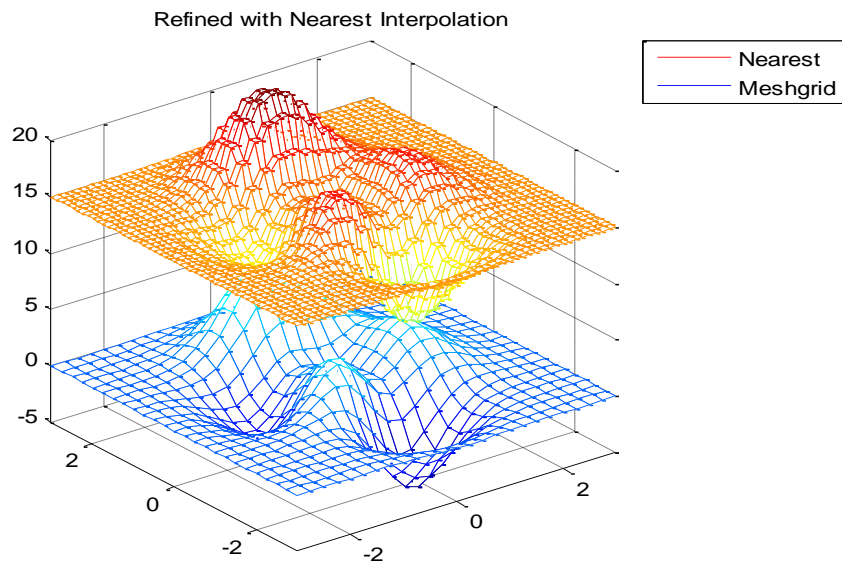


Figure 4.14: Illustration of up-sampling of a 2 dimensional mesh-grid (blue) by a factor of 2 using a nearest neighbour 2D interpolator (red).

However, the nearest neighbour assumption does not permit estimation of new intermediate values, but instead sets the value at the empty location by replicating the pixel value located at the shortest distance and the effect of this is heavy jagged edges. A solution to such jaggedness was achieved through use of another interpolation, known as the bilinear based algorithm, which generates softer images.

4.4.3 Linear Interpolation

Linear interpolation is the simplest method for estimation of missing data without the jagged artefact problem of NN interpolation. In general, a linear interpolator is composed of a series of linear polynomials that are fitted to segments of data and at a minimum the polynomial passes between each pair of data points for curves, or between sets of four points for surfaces (Pan, 2003; Getreuer, 2011).

Bilinear interpolation is an extension of linear interpolation for interpolating images, in which the linear interpolation is applied first in one direction and then the process is performed in another direction for 2D interpolation. In general, for a missing pixel in an image the interpolation is calculated as a weighted average of the attributes of the four surrounding pixels (the closest 2×2 neighbourhood), which are located in diagonal directions from a given pixel, as shown in Figure 4.15 and then appropriate intensity values are assigned to the unknown pixels (as illustrated in Equations 4.10, 4.11 and 4.12).

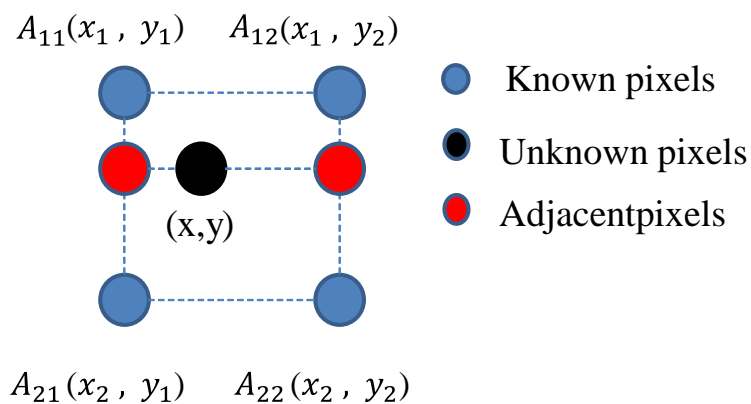


Figure 4.15: Diagram of the linear interpolation algorithm.

To find the unknown value of the function $f(p)$ at the point $p = (x, y)$, linear interpolation is first performed in the x -direction for four known points $A_{11} = (x_1, y_1)$, $A_{12} = (x_1, y_2)$, $A_{21} = (x_2, y_1)$ and $A_{22} = (x_2, y_2)$, as follows:

$$f(R_1) \approx \frac{x_2 - x}{x_2 - x_1} f(A_{11}) + \frac{x - x_1}{x_2 - x_1} f(A_{21}) \quad (4.10)$$

$$f(R_2) \approx \frac{x_2 - x}{x_2 - x_1} f(A_{12}) + \frac{x - x_1}{x_2 - x_1} f(A_{22}) \quad (4.11)$$

Note that $R_1 = (x, y_1)$ and $R_2 = (x, y_2)$. Then, linear interpolation is carried out in the y-direction:

$$f(p) \approx \frac{y_2 - y}{y_2 - y_1} f(R_1) + \frac{y - y_1}{y_2 - y_1} f(R_2) \quad (4.12)$$

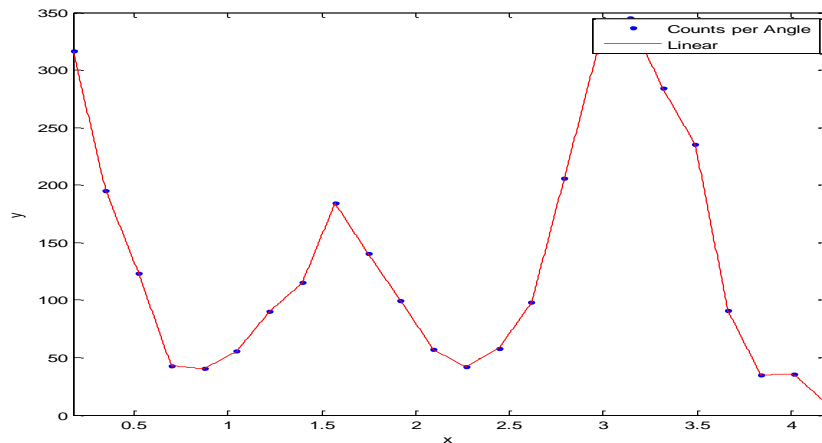


Figure 4.16: Illustration of a one-dimensional linear curve (red) interpolated through a number of known data samples (blue).

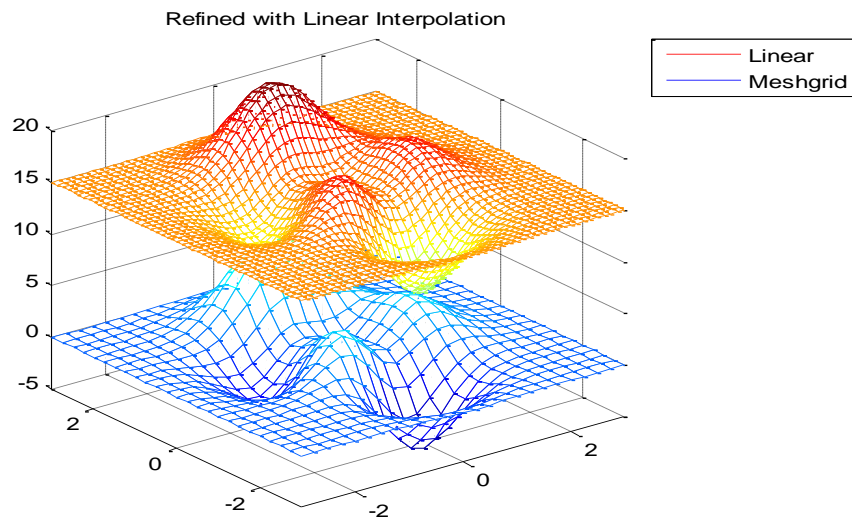


Figure 4.17: Illustration of up-sampling of a 2 dimensional mesh-grid (blue) by a factor of 2 using a linear 2D interpolator (red).

It can be seen from Figure 4.16 and Figure 4.17 that the result for one and two dimensional linear interpolation is improved compared with the NN interpolation,

but there is still some distortion. Table 4.IV shows the result comparisons for bilinear interpolation on Lena, Man, Peppers, Boat and Elaine with a macro block size of 8×8 , where the images were interpolated to the original size of 512×512 having been down-sampled by factors of 8 (64×64). There is an improvement of 0.86 dB in PSNR on average when compare with NN interpolation, but the computational complexity has increased by 2.60 seconds.

Table 4.IV: Performance comparisons for an MB size = 8×8 , on Lena, Man, Peppers, Boat and Elaine for the bilinear interpolation.

Interpolation Method	Performance measure	Images					
		<i>Lena</i>	<i>Man</i>	<i>Peppers</i>	<i>Boat</i>	<i>Elaine</i>	<i>Average</i>
Bilinear	PSNR(dB)	24.91	22.82	25.82	22.70	25.03	24.25
	SSIM	0.6703	0.5178	0.6966	0.5855	0.4345	0.5809
	Time (s)	17.64	17.93	14.22	16.27	16.97	16.60

Although bilinear interpolation is computationally a fast technique, and unlike other interpolation methods, such as bi-cubic interpolation, only considers the closest 2×2 neighborhood values of an unknown pixel to find the appropriate intensity values of that pixel and reduces some of the visual distortion, but it usually yields discontinuities at each point and a smoother method is preferable.

4.4.4 Cubic Interpolation

Cubic interpolation offers continuity between segments by the simplest process amongst all interpolation methods (Pan, 2003; Getreuer, 2011). Bicubic interpolation is an extension to the cubic interpolation technique for interpolating missing data points on a 2D environment. This technique is similar to bilinear interpolation, but it extracts sixteen pieces of information (red and blue dots in Figure 4.18) from the values of at least four known neighbouring points (blue dots in Figure 4.18), in order to make an estimation. Because these sixteen points are at various distances from the unknown pixel, those that are closer are given a higher weighting and thus, have

more influence on the estimation of the missing pixel value in the calculation. As a result, the image is slightly sharper than the one produced by bilinear interpolation.

$$p(x, y) = \sum_{i=0}^3 \sum_{j=0}^3 a_{ij} x^i y^j \quad (4.13)$$

Note that, a_{ij} are constants and x and y are parameters ranging from 0 to 1. The interpolation problem consists of determining the 16 coefficients of a_{ij} . The x , y and xy cross products of these values and the interpolated area can be calculated from the above equation, and the interpolated result area, $p(x, y)$, is continuous.

The interpolation kernel as a solution of the linear system is as follows for each grid cell:

$$s(x) = \begin{cases} 1 - \frac{5}{2} |x|^2 + \frac{3}{2} |x|^3, & |x| \leq 1 \\ 2 - 4 |x| + \frac{5}{2} |x|^2 - \frac{1}{2} |x|^3, & 1 < |x| \leq 2 \\ 0 & \text{otherwise} \end{cases} \quad (4.14)$$

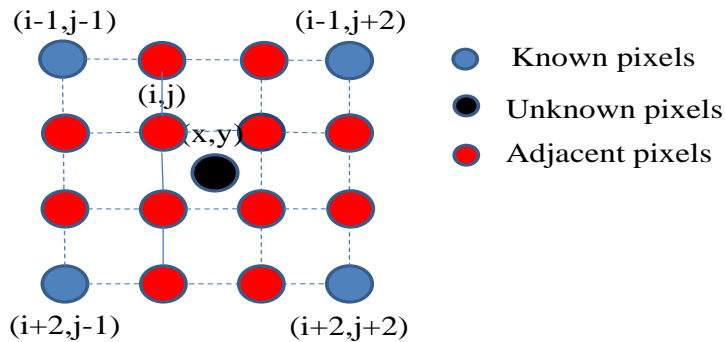


Figure 4.18: Diagram of a cubic interpolation algorithm.

Figure 4.19 and Figure 4.20 demonstrate that the results for one and two dimensional bicubic interpolation are an improvement on NN and linear interpolations. Table 4.V shows the result comparisons for bicubic interpolation on Lena, Man, Peppers, Boat and Elaine, when the macro block size is 8×8 and the images were interpolated to the original size of 512×512 having been down-sampled by factors of 8 (64×64). It can be seen that the performance is higher than for the two previous methods (NN and linear) by 1.5 dB and 0.64 dB, respectively.

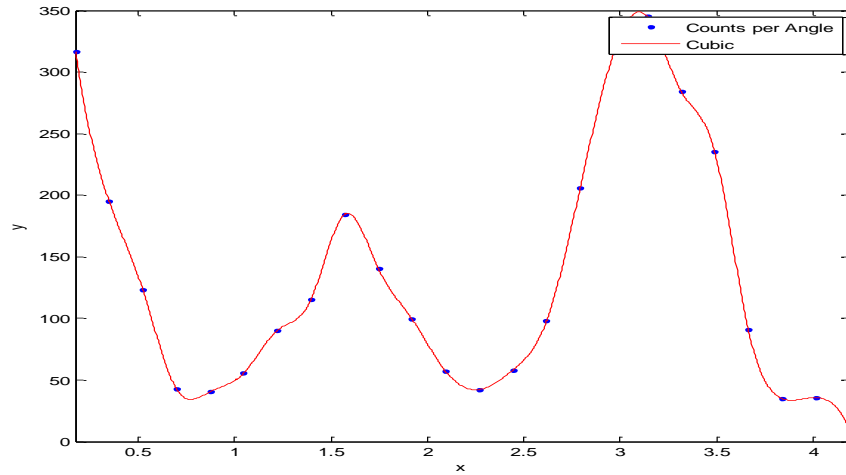


Figure 4.19: Illustration of a one-dimensional cubic curve (red) interpolated through a number of known data samples (blue).

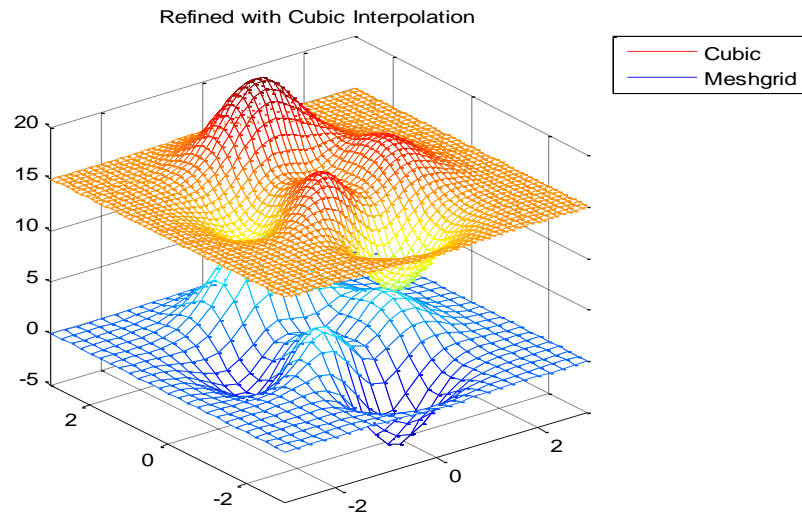


Figure 4.20: Illustration of up-sampling of a 2 dimensional mesh-grid (blue) by a factor of 2 using a cubic 2D interpolator (red).

However, this method is slower than NN and linear interpolation by 5.13s and 2.53s, respectively, because it needs to solve sixteen linear equations, but the result is smoother. Additionally, it can offer true continuity between the segments, which thus makes it a solution for interpolating the surface where higher quality is required and time is not an issue.

Table 4.V: Performance comparisons for an MB size= 8×8, on Lena, Man, Peppers, Boat and Elaine for bicubic interpolation.

Interpolation Method	Performance Measure	Images					
		<i>Lena</i>	<i>Man</i>	<i>Peppers</i>	<i>Boat</i>	<i>Elaine</i>	<i>Average</i>
Bicubic	PSNR(dB)	25.74	23.32	26.59	23.07	25.73	24.89
	SSIM	0.6703	0.5178	0.6966	0.5855	0.4345	0.5809
	Time (s)	20.65	18.61	19.01	19.61	17.77	19.13

4.4.5 Spline Interpolation

A spline function is a higher order (n) piecewise polynomial interpolation involving the use of more surrounding information than a linear interpolator, and as a result it can retain more reconstructed data, which thus means it is much more computationally intensive (Pan, 2003; Getreuer, 2011). For a spline of degree k , each segment is a polynomial of degree k with segment pieces smoothly connected together and the points at which they meet are called knots.

Splines can be uniquely characterised in terms of a b-spline (b stands as basic) expansion, as any spline function of degree k on a number of knots can be expressed as a linear combination of b-splines and the representation is as follows (Pan, 2003):

$$s(x) = \sum_{k \in Z} c(k) \beta^n(x - k) \quad (4.15)$$

where, $c(k)$ are the polynomial functions of order k , and n is the number of control points.

B-splines are most commonly used owing to their computational efficiency. In particular, cubic B-splines offer a good trade-off between the computational cost and the interpolation quality. Owing to the separability property of B-splines, operations on multidimensional data can be carried out by successive processing of one-dimensional (1D) (equation 4.15) data along each dimension. In addition, their multi-resolution property means they are good for making wavelet bases and for multi-scale processing. In general, given these features, many image processing

applications have been designed to use B-splines (Jonic & Sorzano, 2011). They can be implemented efficiently and simply, even though they are mathematically quite complicated.

A one dimension operation can be derived from the Equation (4.15) and the result is as follows:

$$s(x) = \sum_k c(k)\beta(x - k) \quad (4.16)$$

$$s^0(x) = \begin{cases} 1, & -\frac{1}{2} < |x| \leq \frac{1}{2} \\ \frac{1}{2}, & |x| = \frac{1}{2} \\ 0 & \text{otherwise} \end{cases} \quad (4.17)$$

To process an image, the method must consider more than a single dimension. For instance, a 2D spline involves weighting the interpolation in one direction (same as equation 4.15) and then going in another direction. This is defined as follows:

$$s(x, y) = \sum_{k=k_1}^{(k+k_1-1)} \sum_{l=l_1}^{(l+l_1-1)} c(k, l)\beta^n(x - k)\beta^n(y - l) \quad (4.18)$$

2D splines are used in various applications like rotation, zooming, reformatting and resizing.

Figure 4.21 and Figure 4.22 show that there is an improvement for one and two dimensional spline interpolation when compared to the NN, linear and cubic interpolations.

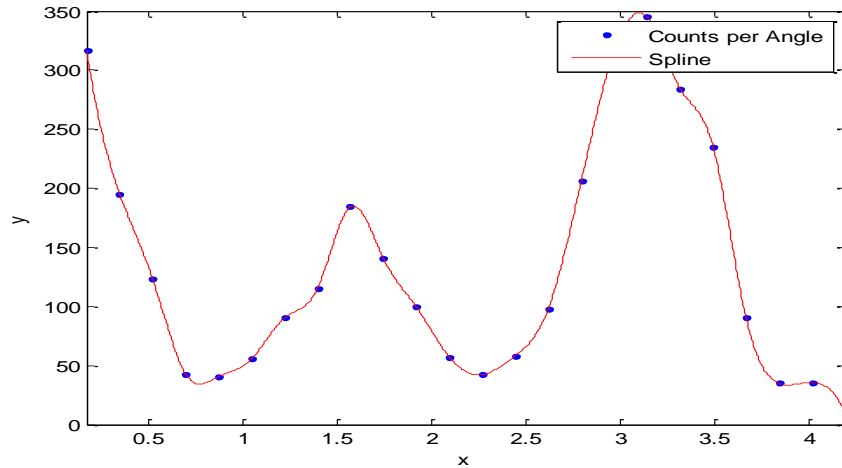


Figure 4.21: Illustration of a one-dimensional spline curve (red) interpolated through a number of known data samples (blue).

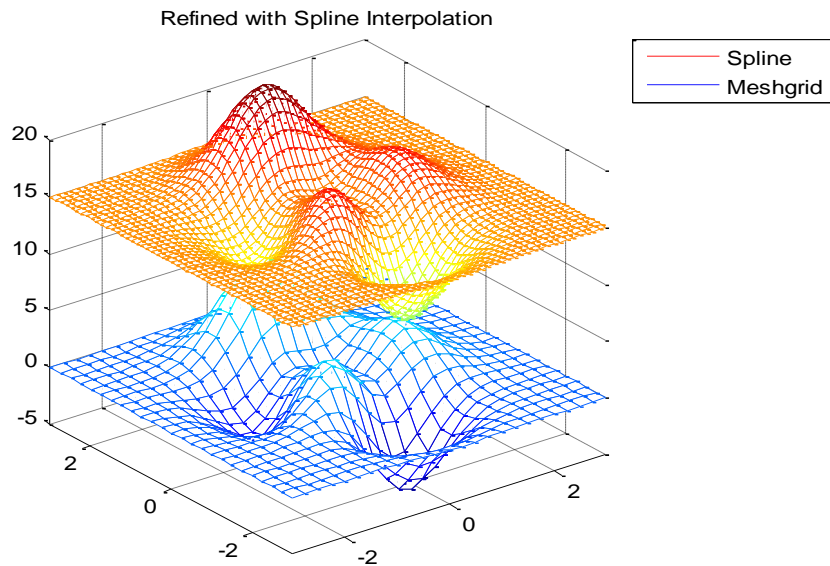


Figure 4.22: Illustration of up-sampling of a 2 dimensional mesh-grid (blue) by a factor of 2 using a B-spline 2D interpolator (red).

Table 4.VI illustrates the result comparisons for spline interpolation on Lena, Man, Peppers, Boat and Elaine when the macro block size is 8×8 and the images are interpolated to the original size of 512×512 having been down-sampled by factors of 8 (64×64). There is an improvement for the spline method by 0.14 dB when compared with the average performance for cubic interpolation. In addition, the spline interpolation surpasses both the NN and linear techniques by 1.64 dB and 0.78 dB, respectively.

Moreover, in the terms of computational complexity, Table 4.VI shows that while the spline method achieves a better outcome for the average processing time over those of cubic by 4.38 s and linear by 1.85 s, this is lower than NN by 0.75 dB.

Table 4.VI: Performance comparisons for an MB size = 8×8, on Lena, Man, Peppers, Boat and Elaine for spline interpolation.

Interpolation Method	Performance measure	Images					
		<i>Lena</i>	<i>Man</i>	<i>Peppers</i>	<i>Boat</i>	<i>Elaine</i>	<i>Average</i>
B-spline	PSNR(dB)	25.99	23.35	26.70	23.16	25.97	25.03
	SSIM	0.6703	0.5178	0.6966	0.5855	0.4345	0.5809
	Time (s)	14.64	13.81	16.31	14.73	14.27	14.75

4.4.6 Comparison Between Different Interpolation Methods

The nearest neighbour and bilinear interpolation methods are both easy to apply, due to their simplicity, but their accuracy is limited and hence, could be insufficient when interpolating high-frequency signals. That is, for these techniques, there is a trade-off between computational complexity and accuracy. NN interpolation is the most efficient of the two in terms of computation time, for bilinear interpolation requires 2.60s more for this. Cubic interpolation requires the most processing time about 19.30s, which is 5.30s more than the computation time of the NN interpolation.

However, the NN generally performs poorly leading to a jagged or blocky appearance. Bilinear interpolation creates a smoother appearance, but the grey levels are changed during computation, thus producing blurring or loss of image resolution. In sum, cubic interpolation provides the best result of these three, but the computation time is very high compared to the rest of these techniques.

It has been observed from experiments, that among all the interpolation techniques the spline method has the greatest approximation, being quite smooth as well as more continuous (Figure 4.23). Moreover, it has the best cost-performance trade off of all the described approaches. In addition, it is the preferred algorithm for multi-

scale approximation, including resizing, pyramids and wavelets as well as for a substantial number of other applications regarding image processing (Jonic & Soriano, 2011).

The experimental results give the guidance to choose the best interpolation algorithm to achieve optimum outcomes, whereby it is essential to limit the interpolation artifacts and this motivates the use of b-splines as the appropriate way to keep them in check without any significant cost penalty.

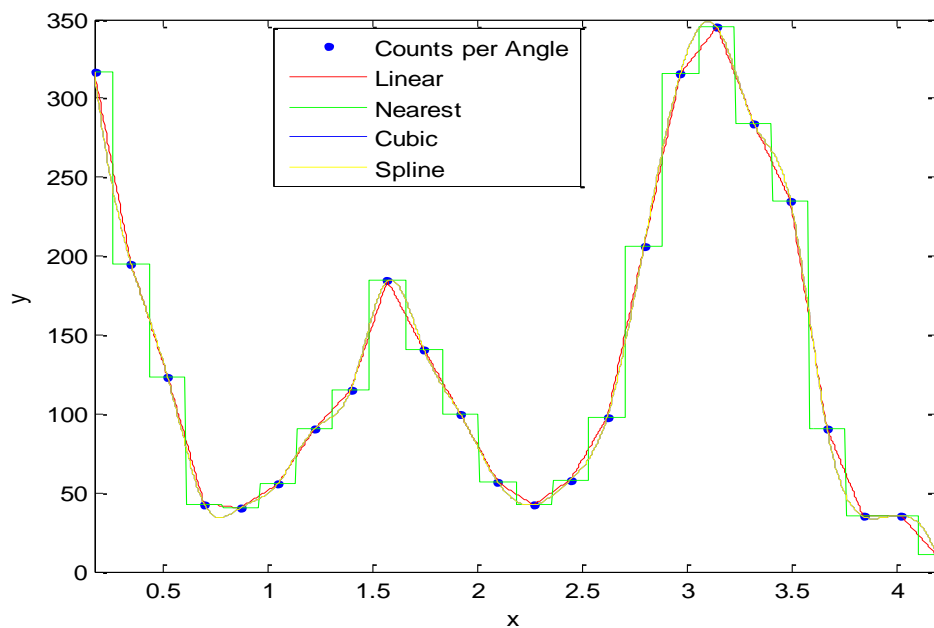


Figure 4.23: Illustration of a one-dimensional NN curve (green), linear curve (red), cubic curve (blue) and spline curve (yellow) interpolated through a number of known samples (blue).

4.5 Evaluation for Regular Loss Pattern

As demonstrated in Figure 4.7, at the last stage of down-sampling, the size of the missing block was reduced from (8×8) or (16×16) to one sample. In order to estimate the missing pixel, two types of estimation are applied: mean estimate and median estimate. The former (Equation 4.19) replaces the missing point by the average of all (8) surrounding pixels (if they are not corrupted), whereas the latter (Equation 4.20) is an edge preserving function, which takes into the account the edge information from all neighbouring pixels.

$$p(i, j) = \text{mean}([A(i - 1:i + 1, j - 1:j + 1, i, j \neq 0)]) \quad (4.19)$$

$$p(i, j) = \text{median}([A(i - 1:i + 1, j - 1:j + 1, i, j \neq 0)]) \quad (4.20)$$

Note that p is the missing pixel at the position (i, j) and $A(i - 1:i + 1, j - 1:j + 1)$ includes all eight surrounding pixels (if they are not corrupted), but not (i, j) itself.

To blend an estimate with the surrounding pixels an interpolator (DCT-based) is applied to the restored image at each level starting from the apex of pyramid, as explained in subsection 4.4.1. Then, the estimate is up-sampled to the next layer by using one of the explained interpolation methods and merged with uncorrupted available pixels, which continues until it reaches the last base level. Various methods of interpolation, which have been seen before, can be used in the up-sampling process, and different results are obtained depending on the different methods, as can be seen in Table 4.VII and Table 4.VIII.

It can be seen (Table 4.VII and 4.VIII) that the median estimator achieves the higher result in comparison with the mean method as it includes edge information in the estimation of the missing pixel. In addition, these tables illustrate that the different techniques that have been used in the up-sampling process provide almost similar results. More specifically, the spline supplies the best outcome the linear and cubic techniques have almost the same result, whilst the nearest neighbour is the worst of all.

The similarity among the results comes from the relatively high correlation that exists among neighbouring image pixels and also, the efficiency of the pyramid DCT structure in capturing the correlations of the image pixels.

Figure 4.24 shows the subjective quality of the recovered images from the original ones with 25% loss rate for a macro block size of 8×8 on Lena, Peppers, Man and Boat for the spline interpolation and some distortion can be seen in all images, especially around the edges. Thus, it is concluded that using interpolation methods alone does not provide an effective recovery result in the missing macro block case.

Table 4.VII: Performance comparisons of different interpolation methods for an MB loss rate of 25% (MB size = 8×8) on Lena, Man, Peppers, Boat and Elaine with a mean estimator at the apex.

Interpolation Method	Performance measure	Images					
		<i>Lena</i>	<i>Man</i>	<i>Peppers</i>	<i>Boat</i>	<i>Elaine</i>	<i>Average</i>
NN	PSNR (dB)	31.21	29.07	31.95	28.68	32.35	30.65
	SSIM	0.9396	0.9006	0.9434	0.9096	0.9217	0.9229
	Time (s)	12.09	12.03	12.38	11.42	11.74	11.93
Bilinear	PSNR (dB)	31.72	29.43	32.46	28.98	32.81	31.08
	SSIM	0.9456	0.9073	0.9493	0.9143	0.9272	0.9287
	Time (s)	11.21	12.47	12.12	12.09	11.79	11.93
Bicubic	PSNR (dB)	31.57	29.32	32.30	28.82	32.69	30.94
	SSIM	0.9446	0.9062	0.9480	0.9129	0.9262	0.9275
	Time (s)	12.22	12.42	12.12	12.31	12.39	12.29
B-spline	PSNR (dB)	31.83	29.63	32.57	29.15	32.94	31.22
	SSIM	0.9463	0.9080	0.9498	0.9151	0.9278	0.9294
	Time (s)	12.12	11.23	11.53	12.49	12.11	11.89

Table 4.VIII: Performance comparisons of different interpolation methods for an MB loss rate of 25%, MB size = 8×8, on Lena, Man, Peppers, Boat and Elaine with the median estimator at the apex.

Interpolation Method	Performance measure	Images					
		<i>Lena</i>	<i>Man</i>	<i>Peppers</i>	<i>Boat</i>	<i>Elaine</i>	<i>Average</i>
NN	PSNR(dB)	31.56	29.27	32.25	28.90	32.74	30.94
	SSIM	0.9440	0.9041	0.9466	0.9132	0.9248	0.9265
	Time (s)	17.10	17.37	21.51	18.72	18.04	18.55
Bilinear	PSNR(dB)	31.89	29.53	32.60	29.09	33.00	31.22
	SSIM	0.9475	0.9089	0.9507	0.9158	0.9286	0.9303
	Time (s)	17.41	17.38	17.69	17.47	17.25	17.44
Bicubic	PSNR(dB)	31.84	29.47	32.53	28.99	33.00	31.16
	SSIM	0.9477	0.9087	0.9503	0.9154	0.9285	0.9301
	Time (s)	19.18	17.74	17.88	19.26	17.70	18.35
B-spline	PSNR(dB)	32.03	29.70	32.70	29.23	33.17	31.36
	SSIM	0.9488	0.9097	0.9523	0.9167	0.9293	0.9313
	Time (s)	17.97	17.92	17.66	18.23	20.76	18.50

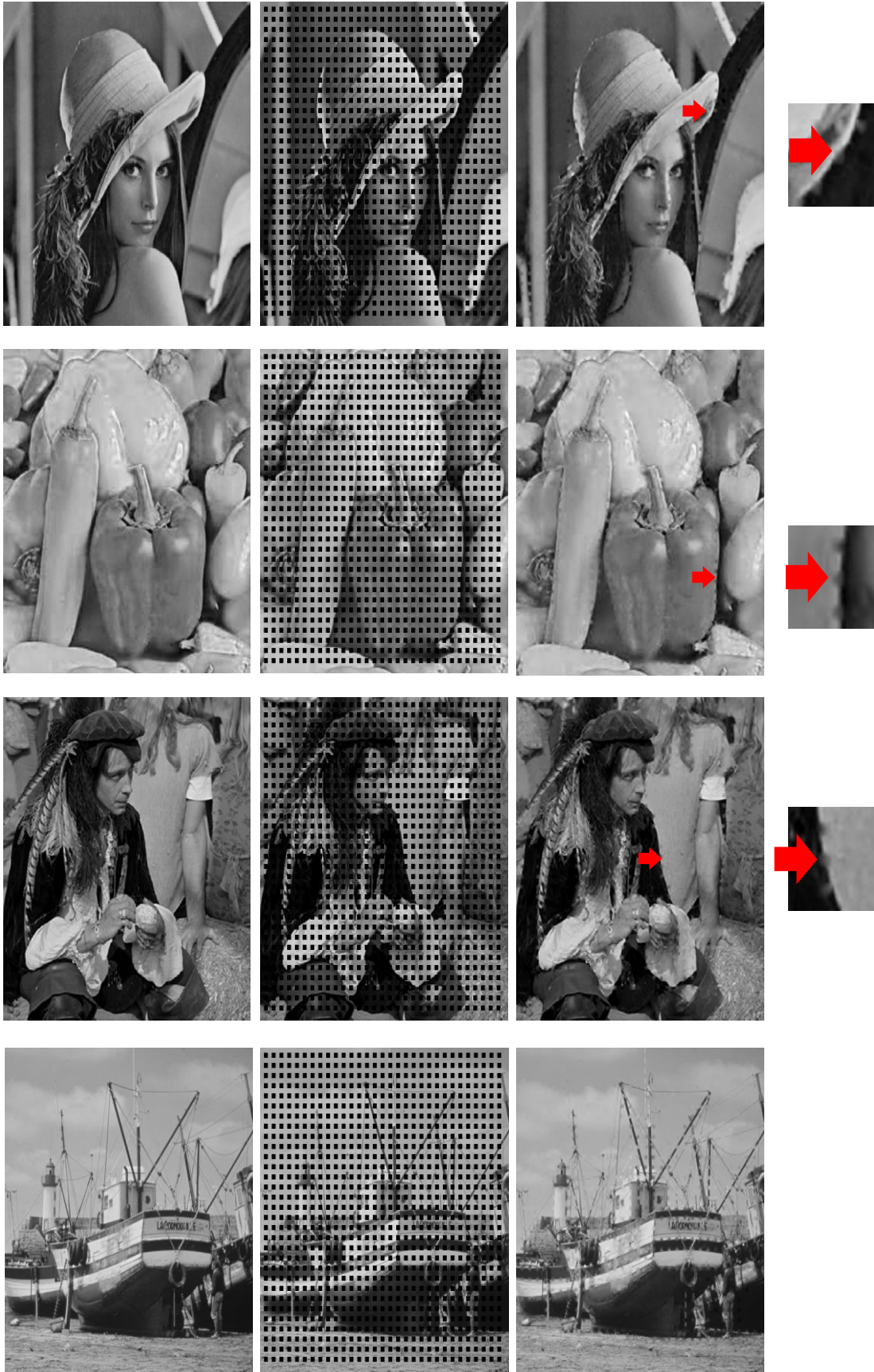


Figure 4.24: From left to right, original images, images with 25% MB (size = 8×8) loss rate, restored and zoomed images on Lena, Peppers, Man and Boat for spline interpolation.

4.6 Conclusion

In this chapter the use of multi-scale techniques which divide the signals into a series of filtered and down-sampled layers of progressively reduced scales and bandwidth has been investigated. Multi-scale transformation into a pyramid layers might facilitate easier interpolation and estimation of missing gaps by creating an information pyramid, where at the apex the missing gap is reduced to one sample only. Consequently, it might facilitate the estimation and reduce the computational complexity.

The main contribution of the work in this chapter has been to establish a baseline for interpolation performance within a DCT pyramid against which the edge-guided interpolation introduced in the following chapter can be compared. The algorithm includes a combination of multi-resolution transforms, different interpolation methods and blending techniques (DCT based interpolation) capable of restoring missing macro blocks.

The impact of using four different interpolation methods for gap estimation was assessed, these being the: nearest neighbour, linear, cubic and spline methods. The interpolators were first compared with regard to their comparative ability to retrieve a down sampled signal without any gap for down-sampling rates 2, 4, 8. The results show that spline interpolation performs best followed by the cubic, linear and nearest neighbour interpolators. However, as the down-sampling rate and hence information loss increases the difference in the performance of the nearest neighbour interpolator (the worst) and the spline interpolator (the best) decreases. Similarly, for the case where there is loss of image macro blocks, then the differences in the performance of various interpolators decrease.

In order to improve the results, the combination of interpolation and post processing, blending, functions were performed with two different estimators (mean and median) at the apex of the pyramid. As expected, the median achieved better results as it is an edge-preserving statistic. The first observation is that all interpolation methods, employed within pyramid estimation, result in reasonably high and very similar values of PSNR of around 31 dB and an SSIM of 0.92. It is proposed that the reasons for the similar results are:

- 1) The relatively high correlation that exists among neighboring image pixels;
- 2) The efficiency of the pyramid DCT structure in capturing the correlations of the image pixels;
- 3) The gap loss, since the experimental result indicates that as the information loss increases the interpolators' performances converge to similar values.

Further work described next includes the use of the interpolation methods in this project as a data-adaptive variety. More specifically, the interpolator coefficients or the regions that they operate on are adapted to the local and global edges of the segments on which they operate.

Chapter 5

Multi-scale Edge-Guided DCT Image Gap Restoration

Chapter 5

5. Multi-scale Edge-Guided DCT Image Gap Restoration

5.1 Introduction

Chapter 5 presents novel edge-guided interpolation methods for image gap restoration through incorporation of edge-based directional interpolation within a multi-scale pyramid transform. Two categories of image edges are proposed and utilised in image gap reconstruction in this research:

- a) The local edges or textures inferred from estimation of the gradients of the neighbouring pixels in various directions and,
- b) The global edges, or boundaries between image objects or segments, inferred using Canny or Sobel edge detectors.

Through a process of pyramid transformation and down-sampling, the image is transformed into a series of progressively reduced size layers until at the pyramid apex the gap size is one pixel. The process is then reversed; at each stage, the missing samples are inferred using estimates of the local and global edges, up-sampled and combined with the uncorrupted samples. For comparison with published works the DCT pyramid is used although a wavelet transform could also be employed. A further justification for using DCT is the fact that most available and widely used applications for image compression coding based on the block coding techniques employ DCT as the transform. Evaluations over a range of images demonstrate that the proposed method improves PSNR and the visual quality compared to a range of published works.

5.2 Local Edges, Textures in Multi-Scale Image Gap Interpolation

The local edges are inferred from the neighbourhood of the missing pixels without the benefit of global edge detectors. The local edges capture the texture information as well as any segment of the global edges that happen to reside within the locality.

At each scale of the reconstruction process, spatial gap concealment interpolates the missing block by using the local edge or texture information obtained from the surrounding available neighbouring pixels. Preserving the texture edges is important for successful error concealment. In this respect, several observations are instructive:

- 1) Along the direction of an edge, the differences of neighbouring pixel values are relatively small;
- 2) Across the direction of an edge the differences of pixel values at the edge discontinuity are relatively large;
- 3) On each side of a gap, the differences of the neighbouring pixel values across an edge are consistent and of similar sign, with the possible exception being where the gap coincides with the end-points of an edge segment.

5.2.1 Local Interpolation of a Single Missing Pixel at the Apex of a Multi-Scale Pyramid

In the proposed pyramid method, edge-based interpolation begins with computing an estimate of the missing pixel at the final level of decomposition, i.e. at the pyramid apex where the missing image gap is reduced to a single sample.

As illustrated in Figure 5.1 a missing pixel at the apex of the pyramid is surrounded by uncorrupted pixels which can be used to define the direction of the edge. Local directional interpolation preserves the following three types of local edges (Figure 5.1):

- 1) Horizontal (H) edges above and below the missing pixels, Figure 5.1.a;
- 2) Vertical (V) edges to the left and right of the missing pixels, Figure 5.1.b;
- 3) Cross (C) edges across four directions, Figure 5.1.c

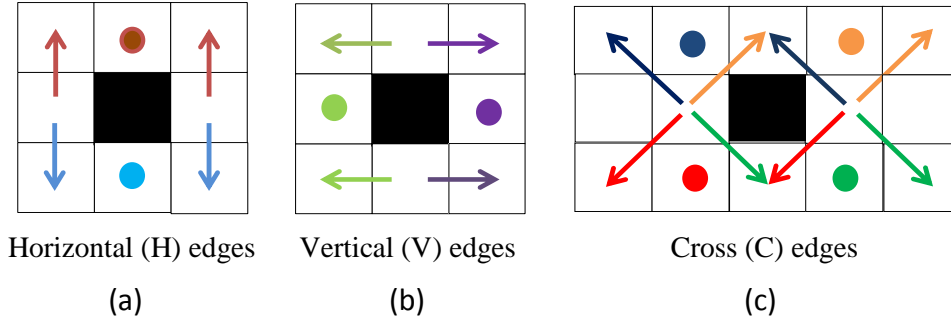


Figure 5.1: Directional interpolation for each missing pixel at the apex of a multi-scale pyramid (in eight possible directions).

The edge-enhanced estimation of the missing sample, $A_{m,n}$, is given by the following Equation 5.1.

$$A_{m,n} = \sum_{H,V,C \in RI} \sum_{k,l} w_{m+k,n+l} (A_{m+k,n+l} + edge_{m+k,n+l}) \quad (5.1)$$

where, $edge(m+k, n+l)$ is a local estimation of the edge obtained separately in each of the horizontal (H), vertical (V) and cross (C) directions, which is shown in Figure 5.1(8 possible directions) and RI is the Region of Interest which for local interpolation, on an un-segmented image, includes information from all neighbouring pixels. The edges along the directions $(m, n) \rightarrow (m+k, n+l)$ are obtained from the average of all the available edges of the same direction in the immediate neighbourhood of the missing sample. For example, at the apex level, where the gap is reduced to one sample, for the horizontal direction, $edge(m-1, n)$ (the brown direction in the Figure 5.1), where the index m denotes horizontal rows and n denotes vertical columns, can be computed simply as:

$$edge_{m-1,n} = \begin{cases} 0 & \text{if } edge_{m-1,n-1} \times edge_{m-1,n+1} < 0 \\ 0.5 (edge_{m-1,n-1} + edge_{m-1,n+1}) & \text{else} \end{cases} \quad (5.2)$$

or a statistical estimate of the average of the majority of neighbouring edges with similar signs can be obtained from a window of length $2l+1$ as:

$$edge_{m-1,n} = stats(edge_{m-1,n-i}) \quad i = -l : +l \quad (5.3)$$

In order to make a weighted estimate consistent with the most distinct neighbourhood edges, the edge combination weights can be expressed as a function of their intensity, as:

$$w_{m+k,n+l} = \frac{edge_{m+k,n+l}}{\sum_{k=-1:1, l=-1:1, k,l \neq 0} edge_{m+k,n+l}} \quad (5.4)$$

Note that $\sum_{k=-1:1, l=-1:1, k,l \neq 0} w_{m+k,n+l} = 1$

5.2.2 Local Block Interpolation at $[N \times N]$ Pyramid Scale

After interpolation of the apex sample, at the subsequent stages of interpolation, for blocks of size $N \times N$ a strategy similar to that described above is used. Starting from the outer boundaries of the macro block, the missing pixels are progressively replaced towards the centre, while the local edge-guided interpolation methodology aims to obtain estimates that are consistent with the neighbouring edges in each of the horizontal, vertical and cross directions.

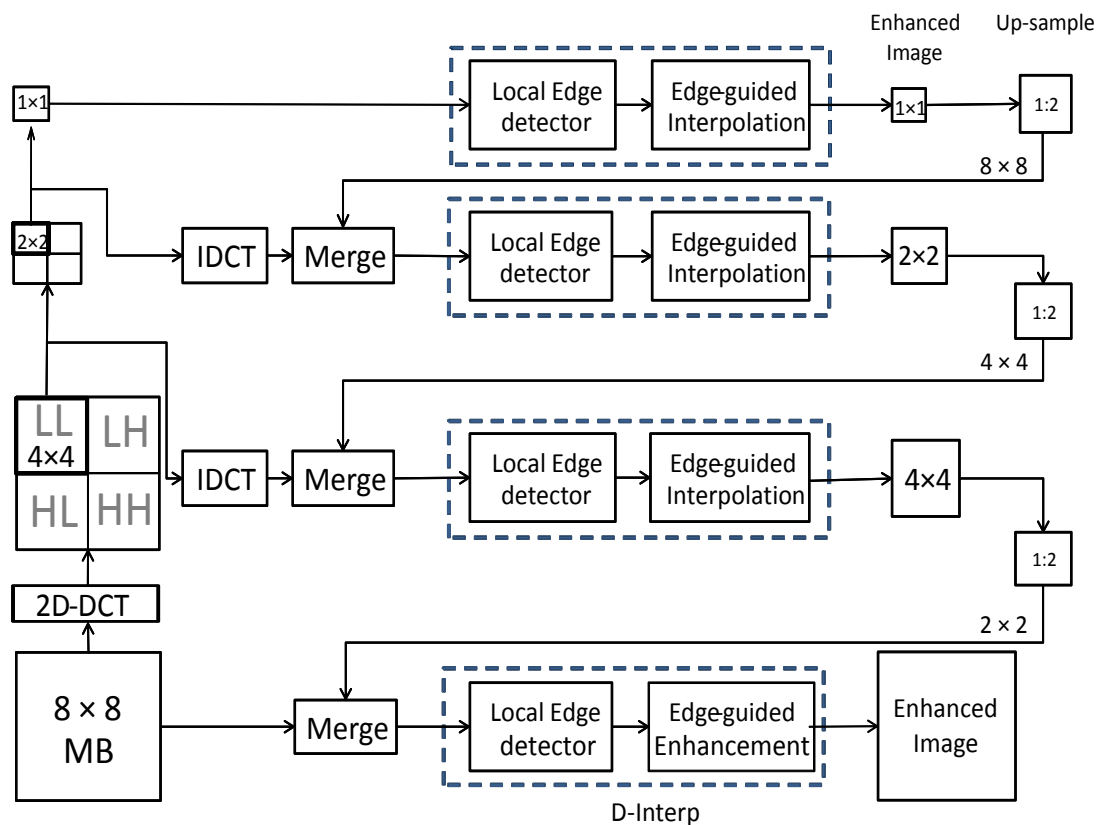


Figure 5.2: Diagram of the process in the local edge-guided interpolation.

This estimate is an edge-weighted mean of the available or already estimated neighbouring pixels with consistent edges. At the successive levels where an $N \times N$

interpolated block replaces a gap, local directional edge-guided interpolation is used to fit the missing blocks with the edge patterns of the available neighbouring pixels. Figure 5.2 illustrates the process in local edge-guided interpolation.

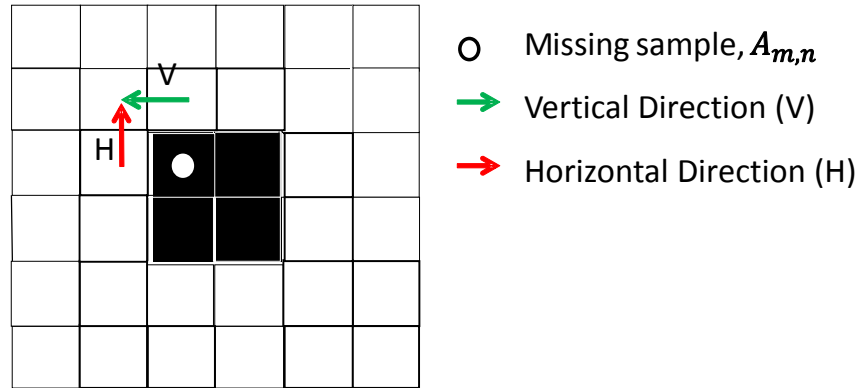


Figure 5.3: Interpolation using the neighbourhood edges, inferred from the pixels available (on three sides of the missing pixel) in two directions: horizontal (H) (red arrow) and vertical (V) (green arrow).

Figure 5.3 is an illustrative example of the method for edge estimation for a 2×2 block, one possible way of using neighbourhood edges inferred from the available pixels (on three sides of the missing pixel), in two directions (horizontal (H) and vertical (V)). In addition, the estimated pixel itself from the previous steps is employed in order to compute the unknown pixel value.

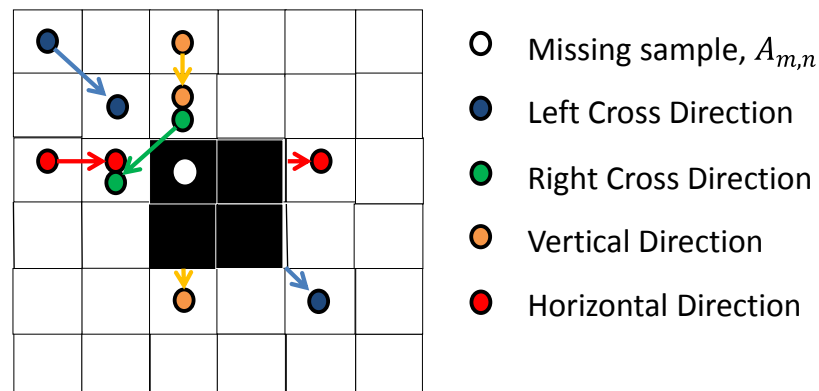


Figure 5.4: The interpolation inferred from four possible directions (represented by four different colours) combined with the pixel estimate (white dot).

A more complete form of inclusion of neighbourhood edge information in the inference process is illustrated in Figure 5.4. That is, interpolation inferred from four possible directions combined with the pixel obtained from the previous stage is used to compute a new estimate of the missing pixel at the block of size $N \times N$. As

shown in Figure 5.4, three pixels are involved in each direction, comprising two immediate neighbouring pixels and a third which is the first available pixel after the gap in the same direction. However, there is an exception for each pixel in one of the cross directions, which just includes two instantly available pixels (right cross in Figure 5.4). As demonstrated in Figure 5.4, which refers to the missing block of size 2×2 , four types of edges are applied to estimate the missing sample, $A_{m,n}$, through a unique order:

- 1) Horizontal (H) edges above and below the missing pixels, Figure 5.4. (Red dots);
- 2) Vertical (V) edges to the left and right of the missing pixels, Figure 5.4. (Amber dots);
- 3) Right Cross (RC) edges, Figure 5.4. (Green dots);
- 4) Left Cross (LC) edges, Figure 5.4. (Blue dots).

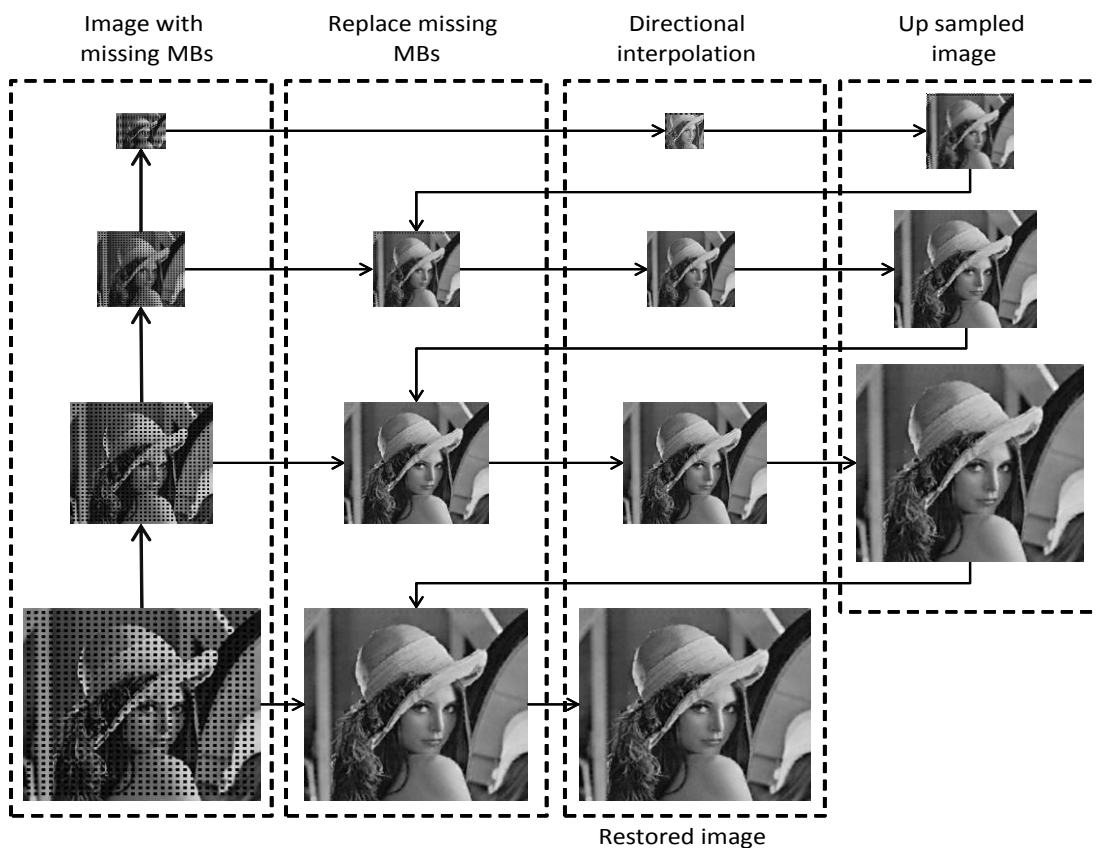


Figure 5.5: Illustration of pyramid DCT decomposition of Lena. A 8×8 gap at the base is transformed to a single missing pixel at the apex and directional interpolation composed of local edge information is used to infer missing pixel values. After up-sampling the estimates are used to replace the gap at a lower pyramid layer and the process is continued to the pyramid base.

The edge-enhanced estimation of the missing sample, $A_{m,n}$, is given by using the Equation 5.1 with a different RI. In addition for each pixel the distance normalised weight is computed so as to obtain a more accurate result (Equation 5.5).

$$A(m, n) = \sum_{pq} \alpha_{pq} A(m + p, n + q) \quad (5.5)$$

$$\text{where } \alpha_{pq} = \frac{d_{pq}}{\sum d_{pq}} \text{ and } d_{pq} = \sqrt{p^2 + q^2}$$

where m, n, p, q denote the position indices of the pixels, d_{pq} is the Euclidean norm or distance, α_{pq} is the distance normalised weights and $\sum_{p,q} \alpha_{pq} = 1$.

Figure 5.5 shows the objective result of pyramid DCT decomposition of Lena with a missing MB of size 8×8 , using the local edge-guided interpolation.

5.2.2.1 Post-Processing; Blending, Block Estimates with Surroundings

As a further enhancement processing step, to blend the estimate of a block within its surrounding texture a DCT based mixing akin to a form of data dependent low-pass filtering is used. The method for a missing block of size 2×2 can be described as follows and is also demonstrated in Figure 5.6.

- Step 1: Take a 4×4 block that includes the 2×2 estimate at its core centre.
- Step 2: Perform a 4×4 DCT, retain the low frequency 2×2 or 3×3 subset and set the remaining coefficients to zero, this is low-pass filtering.
- Step 3: IDCT the modified 4×4 set.
- Step 4: Retain the core 2×2 as the blended estimate.

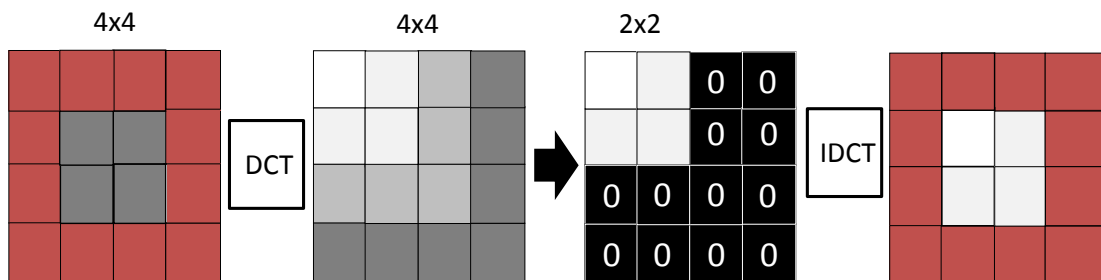


Figure 5.6: Post-processing: Blending of block estimates with surrounding pixels for a missing block of size 2×2 .

5.2.3 Experimental Result for Local Edge-Guided Interpolation

Regarding performance evaluation results, the proposed algorithm has been tested on a number of standard test images, including: Lena, Peppers, Man and Foreman. The image sizes are 512×512 pixels, with each grey-scale or one of the primary colours represented by 8 bits per pixel in unsigned integer format, with a range of 0-255. The size of the missing macro block has been set to 8×8 and 16×16 pixels. Three typical missing macro blocks are considered:

- Regular missing macro block: at 25% loss rate for 8×8 missing macro blocks (Figure 5.7) ;
- Regular missing macro block: at approximately 10% loss rate for 16×16 missing macro blocks (Figure 5.8) ;
- Random (mixed) missing macro blocks: with the loss rate set to 10% for 8×8 missing macro blocks (Figure 5.9).

The choice of the percentage loss is guided by the desire to compare the results illustrated in Tables 5.I, 5.II and 5.IV with available well-known ones reported in the literature.

The performance measurement criteria used for assessment of the quality of image recovery is the widely employed Peak-Signal-to-Noise-Ratio (PSNR), which is defined as:

$$PSNR = 20 \log_{10} \frac{MAX_I}{RMSE} \quad \text{dB} \quad (5.6)$$

The aim of error concealment is to minimise the root mean squared (RMSE) error metric, and thus maximise the *PSNR*. These metrics are straightforward and easy to evaluate. Where $MAX_I = 255$ for a pixel value represented in unsigned integer format and the RMSE function is defined as:

$$RMSE = \sqrt{\frac{1}{N} \sum_{domain} (A(m,n) - Ar(m,n))^2} \quad (5.7)$$

where, $A(m,n)$ and $Ar(m,n)$ are the original and recovered pixel value, respectively. The domain, over which the RMSE is calculated might include only the

missing samples or it, alternatively, can include the entire image samples composed of those missing as well as those available, and N is the total number of samples used in calculation.

5.2.3.1 Evaluation for Regular Loss Pattern

The local edge-guided interpolation method has been applied to the images of Lena, Peppers, Man and Foreman, as shown in Figures 5.7 and 5.8. The PSNR results are compared to a set of seventeen published works representing a number of methods that have employed Bayesian and/or edge information for the recovery of regular lost macro blocks. The results are displayed in two different tables. Table 5.I represents comparison with the published results, where the PSNRs are averaged over the whole image including the available samples and Table 5.II represents comparison with published results where the PSNRs are averaged over the missing pixels only.

Table 5.I illustrates the performance of several methods (values are taken from Kim, Koo & Jeong, 2006; Liu *et al.*, 2014). As displayed in Table 5.I, the proposed method performs better than all the alternatives considered and there is an improvement of 0.46 dB compared with the best average performance of any other method, which is that of Kim, Koo & Jeong (2006) when the PSNRs were computed from the whole image.

In Liu *et al.* (2015) the result is provided only for Lena and it is over by 1 dB. The lowest outcome among the rest (Ancis & Giusto, 1999) is at 27.35 and the proposed method is above this by 6.61 dB. There is almost the same range of results for Sun & Kwok (1995), Hemami & Meng (1995), Shirani, Kossentini, & Ward (2000), and Alkachouh & Bellanger (2000) at 29.07, 30.28, 30.44 and 30.75, respectively, and as such, they are approximately 3.50 dB lower than the proposed method of the current study. Park *et al.* (2005) achieved an outcome of 32.90 dB, but the proposed method surpasses it by 1.06 dB. There is a higher average result for the local edge-guided interpolation, as can be seen in the table below, when compared with the best performance (Kim, Koo & Jeong, 2006) by 0.39 dB. Despite the proposed method obtaining a better outcome for the Man and Peppers images compared with Kim, Koo & Jeong (2006) by 0.86 dB and 0.53 dB, respectively, the performance for

Lena's image with respect to theirs was slightly lower by 0.20 dB, as this method just utilises the local information.

Table 5.I: Performance comparisons for MB loss rate of 25%, MB size=8×8, with the PSNRs calculated over the whole image for Lena, Man and Peppers for local edge interpolation.

Methods	PSNR (dB)			
	<i>Lena</i>	<i>Man</i>	<i>Peppers</i>	<i>Average</i>
Ancis (Ancis & Giusto, 1999)	28.68	25.47	27.92	27.35
Sun (Sun & Kwok, 1995)	29.99	27.25	29.97	29.07
Shirani (Shirani, Kossentini, & Ward, 2000)	31.69	27.44	31.72	30.28
Hemami (Hemami & Meng, 1995)	31.86	27.65	31.83	30.44
Alkachouh (Alkachouh & Bellanger, 2000)	31.57	27.94	32.76	30.75
Park (Park <i>et al.</i> , 2005)	34.65	29.87	34.20	32.90
Kim (Kim, Koo & Jeong (2006)	34.91	30.62	35.18	33.57
Liu (Liu <i>et al.</i> , 2014)	35.71	--	--	--
DCT Local-edge	34.71	31.48	35.71	33.96

In Table 5.II (values are taken from Li & Orchard (2002), Zhao *et al.* (2005) and Zhai *et al.* (2010)) the PSNR is calculated for the region of missing sample blocks only. It can be seen that Agrafiotis, Bull & Canagarajah's (2006) outcome provides the worst performance, with proposed method achieving an improvement of 5.44 dB over theirs. Li & Orchard (2002) introduced Sequential Error-Concealment, such that the previously recovered pixels can be used in the recovery process afterwards, but compared with the local edge-guided interpolation their result is lower by 0.16 dB. Jung, Chang & Lee (1994), Park *et al.* (2005) and Zhao *et al.* (2005) obtained similar results, whereas the proposed method's outcomes surpassed theirs by 2.07 dB, 2.41 dB and 1.48 dB, respectively. However, Zeng & Liu (1995), who made use of the local geometric information extracted from the surrounding blocks to interpolate the missing pixels, achieved a PSNR at 27.43 and 0.98 dB lower than the proposed method. The rest of the techniques obtained approximately the same results, ranging from Alkachouh & Bellanger (2000) at 24.00 dB to Hsia (2004) at 25.14 dB and

local edge-guided interpolation provides a significantly higher outcome when compared with these. The best performance among all the compared methods is that Zhai *et al.* (2010) at 28.51 dB. They employed a Bayesian estimation approach with a discrete cosine transform, obtaining a 0.10 dB greater outcome than for the proposed method.

Table 5.II: Performance comparisons for MB loss rate of 25%, MB size= 8×8, with PSNRs calculated just for the region of the missing block on Lena for local edge interpolation.

Gap Estimation Methods	Image Lena
	PSNR (dB)
Zhai (Zhai <i>et al.</i> , 2010)	28.51
Agrafiotis (Agrafiotis, Bull & Canagarajah, 2006)	22.97
Park (Park <i>et al.</i> , 2005)	26.00
Zhai (Zhai <i>et al.</i> , 2008)	28.11
Zeng (Zeng & Liu 1995)	27.43
Li (Li & Orchard, 2002)	28.25
Jung (Jung, Chang & Lee, 1994)	26.34
Wang(Wang, Chang & Shaw, 1993)	24.70
Alkachouh (Alkachouh & Bellanger, 2000)	24.00
Shirani (Shirani, Kossentini, & Ward, 2000)	24.50
Sun (Sun & Kwok, 1995)	24.70
Hsia (Hsia, 2004)	25.14
Zhao (Zhao <i>et al.</i> , 2005)	26.93
DCT Local-edge	28.41

5.2.3.2 Evaluation for Random Loss Pattern

In order to evaluate the performance result of the local edge-guided interpolation for the random (mixed) block loss, the method has been applied to the Lena, Peppers, Man and Foreman images, as shown in Figure 5.9, and compared with the published methods (Li & Orchard, 2002 ; Zhai *et al.*, 2010). In the mixed model of packet loss there is no specific pattern of loss, and in this case two or more lost macro blocks could be adjacent.

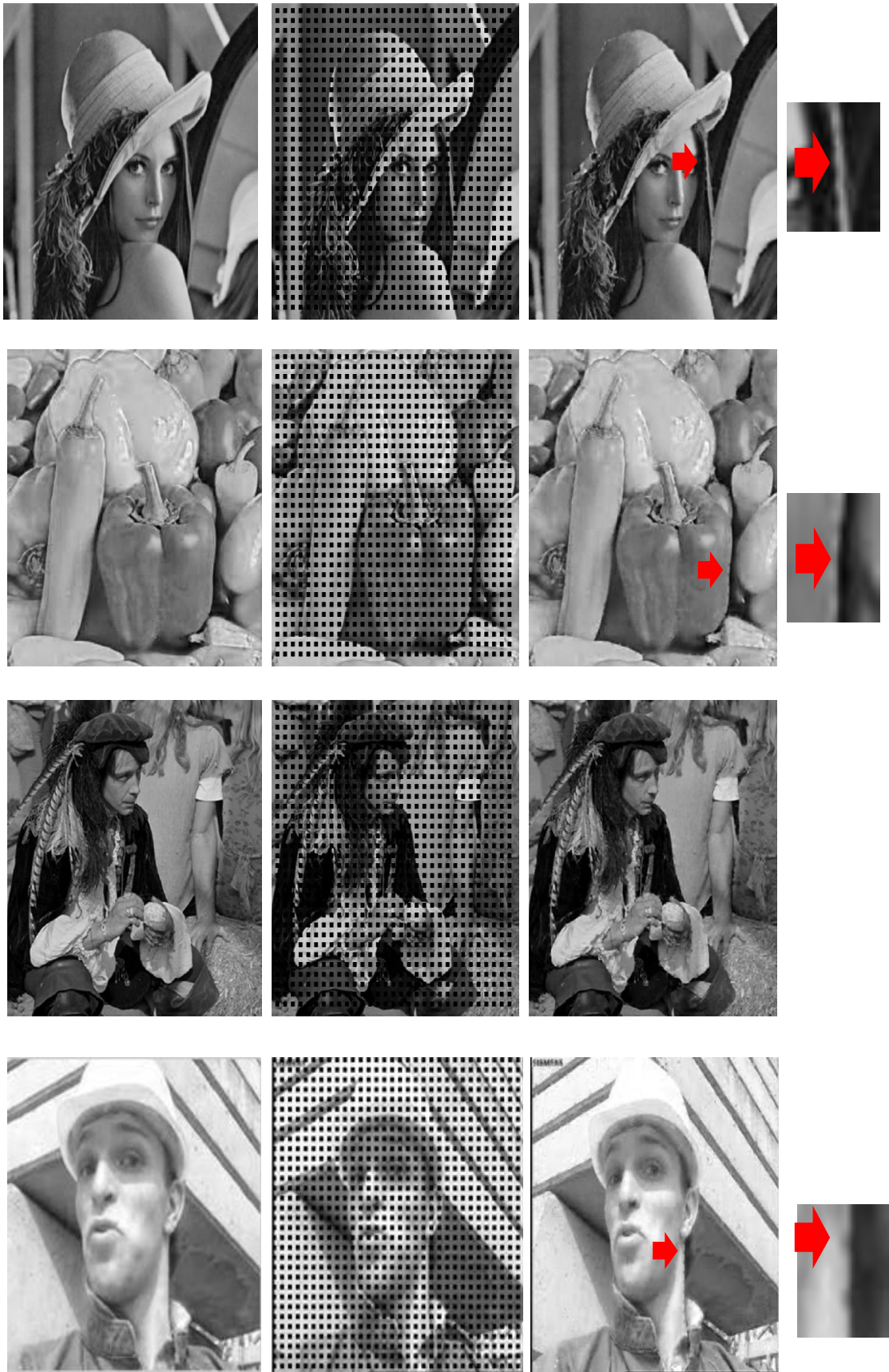


Figure 5.7: From left to right; the original images, the image with 25% MB loss (8×8 isolated block loss) and the restored images for Lena, Pepper, Man and Foreman.

In order to have a more reliable test result the variance and standard deviation (SD) (Equation 5.8) are calculated for a number of simulations so as to show that the mean amount of PSNR is in the range.

$$SD = \sqrt{\frac{1}{N} \sum_{i=1}^N (PSNR_{mean} - PSNR_i)^2} \quad (5.8)$$

where, N is a number of iterations, $PSNR_{mean}$ illustrates a mean of all N PSNRs and $PSNR_i$ represents the PSNR of the i^{th} iteration.

Table 5.III: Performance comparisons for a random MB loss rate of 10% (MB size = 8×8) on Lena with ten steps of iteration.

Image \ Result	Lena									
	1	2	3	4	5	6	7	8	9	10
PSNR (dB)	31.51	31.32	30.98	31.19	31.40	30.64	30.37	31.00	31.21	30.86

To start, the mean of ten iterations of PSNR values is computed, afterwards the variance and standard variance are calculated using Equations 5.9 and 5.10. The small variance indicates (Equation 5.10) that the data points tend to be very close to the mean and hence, to each other. Then, the mean value is used to compare the result with published state of the art methods.

$$Mean = \sum_{i=1}^{10} PSNR_i = \frac{310.48}{10} = 31.048 \approx 31.05 \quad (5.9)$$

$$Variance = \frac{1}{10} \sum_{i=1}^{10} (31.05 - PSNR_i)^2 = \frac{1.1262}{10} = 0.11262 \quad (5.10)$$

$$SD = \sqrt{Variance} = \sqrt{0.11262} = 0.3355 \quad (5.11)$$

Table 5.III (values are taken from (Li & Orchard, 2002; Zhai *et al.*, 2010)) includes the results for four published techniques and the proposed method (average of ten iterations). The PSNRs are averaged over the region of missing sample blocks only and hence, do not include the available samples.

Table 5.IV: Performance comparisons for a random MB loss rate of 10%, MB size= 8×8, with PSNRs calculated just for the region of the missing block for Lena.

Methods	Image Lena
	PSNR (dB)
Zhai (Zhai <i>et al.</i> , 2010)	28.13
Zhai (Zhai <i>et al.</i> , 2008)	27.65
Zeng (Zeng & Liu, 1995)	26.60
Li (Li & Orchard, 2002)	27.38
DCT Local-edge	31.05

As can be seen in the table above, the proposed method is better than all published works. The best result among the rest is that of Zhai *et al.* (2010), which achieved 2.92 dB less when compared with the proposed scheme. Two other techniques have almost the same performance results, being under the current method outcome by 3.4 dB in Zhai *et al.*'s (2008) case and 3.67 dB in that of Li & Orchard (2002). The worst result is for Zeng & Liu (1995) at 26.60 dB, and the proposed method surpassed it by 4.45 dB.

Figure 5.8 shows the subjective quality of the recovered image and some distortion can be seen especially around the sharp edges. Thus, it is concluded that local edge-guided interpolation is not able to provide an effective result in the random missing macro block case and a more advanced method is required.



Figure 5.8: From left to right: the original Lena, Peppers, Man and Foreman images, the images with 10% random missing MBs and the restored image for local edge interpolation.

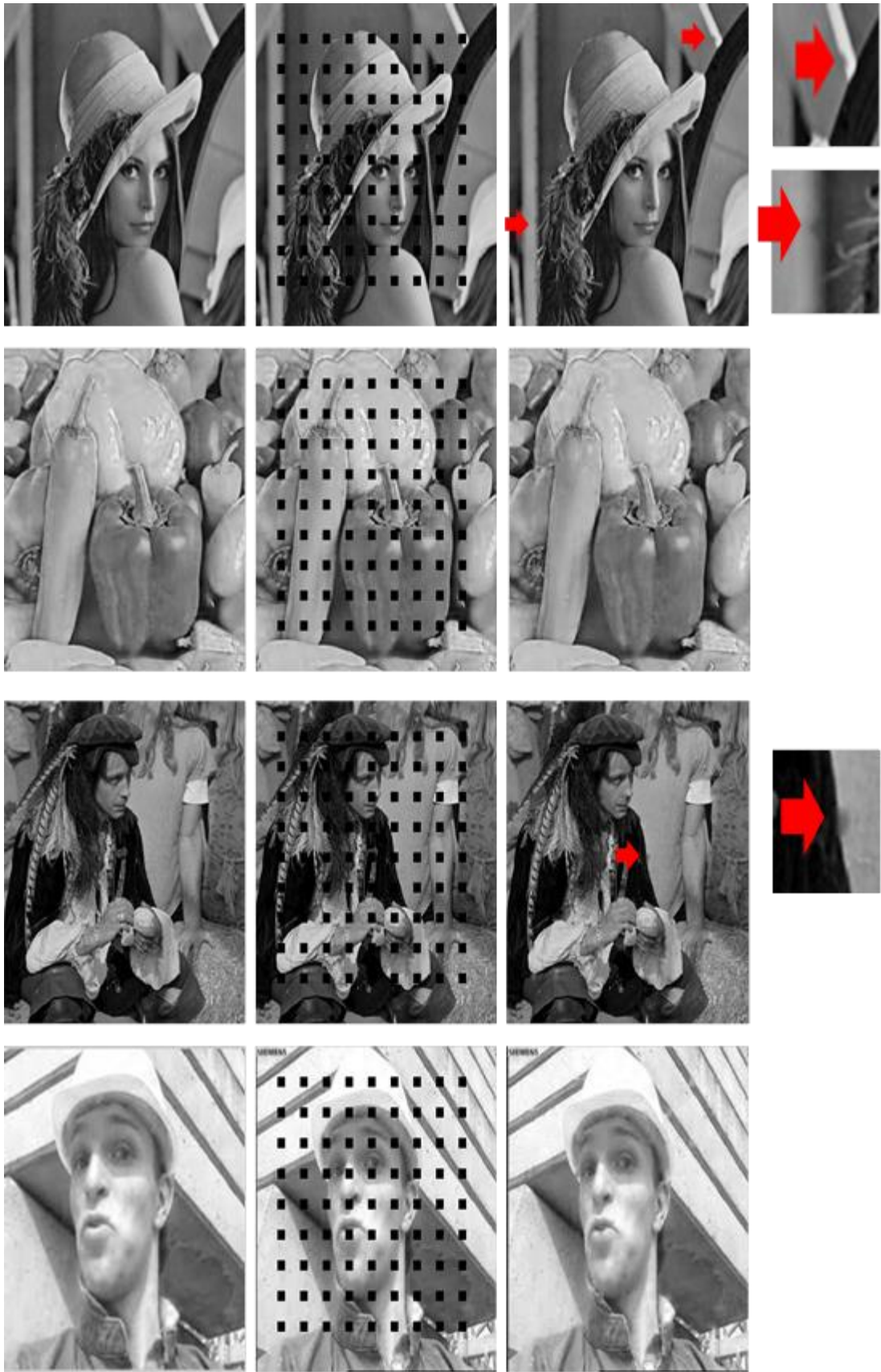


Figure 5.9: From left to right: the original images, corrupted images (16×16 isolated block loss), the restored and zoomed in images for Lena, Peppers, Man and Foreman for local edge-guided interpolation.

5.2.3.3 Evaluation for 16×16 Regular Loss Pattern

A block loss size of 16×16 pixels is also applied to the proposed algorithm for further evaluation, for which a 5-level DCT pyramid is used and Figure 5.9 demonstrates the subjective performance. In addition, Table 5.V shows the PSNR comparison between the proposed algorithm and previous works.

Table 5.V: Performance comparisons for MB loss size= 16×16 , with PSNRs calculated over the whole image for Lena, Man and Peppers with local edge interpolation

Methods	PSNR (dB)		
	<i>Lena</i>	<i>Peppers</i>	<i>Average</i>
Salama (Salama, Shroff & Delp, 1998)	35.01	34.71	34.86
Wang (Wang, Zhu & Shaw, 1993)	35.43	35.07	35.25
Sun (Sun & Kwok, 1995)	34.95	33.21	34.08
Park (Park <i>et al.</i> 2005)	35.98	35.50	35.74
Li (Li & Orchard, 2002)	37.48	38.27	37.87
Kim (Kim, Koo & Jeong, 2006)	37.37	38.95	38.16
DCT Local-edge	35.61	36.22	35.91

Table 5.V contains objective comparisons between the local edge-guided interpolation method and six published works on the Lena and Peppers images. From the simulation result, while the average for the proposed scheme has a PSNR over those of Salama, Shroff & Delp (1998), by 1.05 dB, Wang, Zhu & Shaw, (1993) by 0.66 dB, Sun & Kwok (1995) by 1.83 dB and Park *et al.* (2005), by 0.17 dB, it is not as effective as the other two techniques. The best result of all is that of Kim, Jasung Koo & Jeong (2006), which exceeds that of the local edge-guided algorithm by 2.25 dB and Li & Orchard's (2002) is higher by 1.96 dB.

It can be seen from Figure 5.9, the test result is similar to the 8×8 model, and it can also be observed (in addition to Figure 5.7 and 5.8) that although there is a significant improvement in the local areas compared with the results of Chapter 4, severe losses are not successfully reconstructed in the edge regions. Moreover, there

is some distortion along the edges, which is indicated by a red arrow in the restored Lena in Figure 5.9. Hence, it is evident that a further process, such as a global-edge guided interpolation, is required to recover the edge regions that have been neglected by the current approach.

5.3 Global Edge-Guided Image Gap Interpolation

From the above analysis and evaluation it is observed that preserving the *local* edges mitigates blurring distortions of textures and provides improved interpolation at a local texture level, in particular at the boundaries of the available and missing samples. For further improvement where the missing blocks contain significant edges, such as major boundaries between segments and objects, the *global* edge information, not necessarily evident within the lost macro blocks, needs to be utilised.

The detection of edges is a necessary process in the proposed method. The edge is defined as a sharp change in intensity therefore, edge detection applications are used in order to identify the presence and location of these intensity step changes. Edge detection significantly reduces the amount of data and filters out unwanted or insignificant information, thereby providing the significant information and specifying edges within an image.

The proposed method for global edge-guided image gap interpolation is similar to the local edge-guided interpolation, but it has an additional process at each level. First, the boundaries in the image are identified through the edge detection and then, more data can be recovered by using that information. The algorithm is as follows (Figure 5.10):

- 1) Decompose image macro-blocks into a DCT pyramid structure, with the apex of the pyramid representing the last stage of down-sampling (where each MB of size 8×8 is reduced to only one pixel after three stages and for an MB of size 16×16 is reduced to only one pixel after four stages);

- 2) Starting from the apex of the DCT pyramid interpolate the decimated (down-sampled) gap using the local edge information from the neighbouring pixel, as described in section 5.2;
- 3) Use an edge detector, such as Sobel or Canny, to track the global edges in the interpolated images;
- 4) Enhance the interpolated gap estimation using the global edge information,
- 5) Up-sample the enhanced interpolated image via zero-padded inverse 2D-DCT and combine/merge with the available received samples of the same layer of up-sampling;
- 6) Go to step (1) and repeat the process for each intermediate stage of up-sampling.

The details of these sub-processes are illustrated in Figure 5.10.

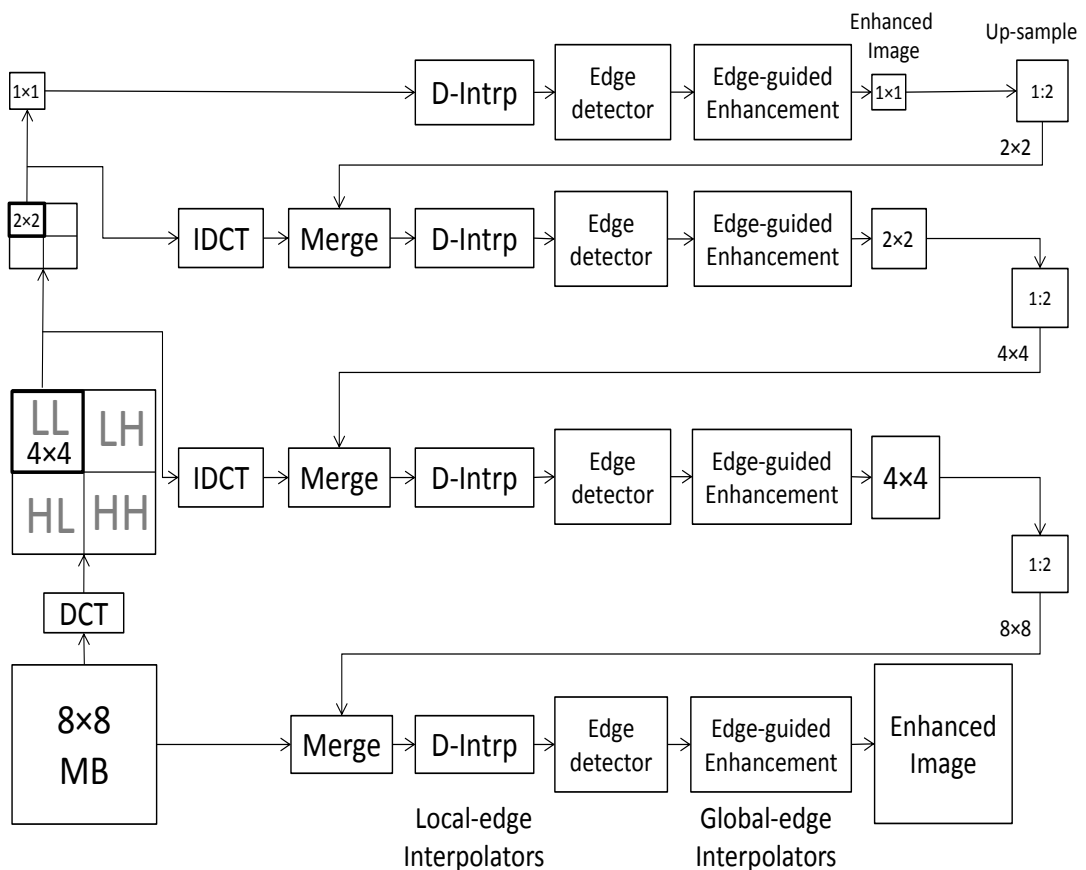


Figure 5.10: Three-stage DCT pyramid image decomposition (D-Interp = Directional Interpolation).

5.3.1 Edge Detection

The global edges are used to avoid or mitigate the problem of blurred/smeared interpolation across the significant edges, which is the main cause of large interpolation errors and visible distortions in image restoration. With the availability of the boundary traces of the edges, it is possible to segment the pixels (Figure 5.11) within and in the neighbourhood of missing blocks as well as to confine the available samples used for interpolation of a missing sample within a relatively homogeneous region on each side of the edge or onto the edge itself, as required.

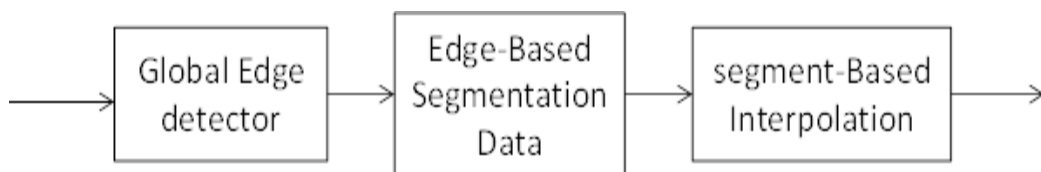


Figure 5.11: Segmentation of pixels with a global edge detector.

Figure 5.12 illustrates an edge-based segmentation of an image into two broadly homogenous contrasting texture areas. The dark sections are on either side of the edges or lie within closely-spaced edges, whereas the light areas represent the pixels on the edges. Brighter pixels demonstrate the stronger edges within an image.



Figure 5.12: From left to right an estimated image of Lena after local edge-guided interpolation in the last stage, and edge-based segmentation of an image with the Sobel edge detector.

Note regarding Figure 5.12 that the global edge-guided interpolation is performed after local edge interpolation, in order to mitigate the impact of the missing samples

on the edge detection. For estimation of the main edges in the image, the applications of two popular edge detection methods were investigated, namely, the Sobel and the Canny edge detectors, which allowed for comparison so as to identify that which provides the best results.

Figure 5.13 illustrates the proposed multi-scale method of restoration for the corrupted image of Lena with 25% macro block loss (size of 8×8). Note that in the case of having an macro block of size 16×16 , the method consists of four stage of down/up-sampling. The figure shows that after applying global edge information the result is improved at each stage of the process.

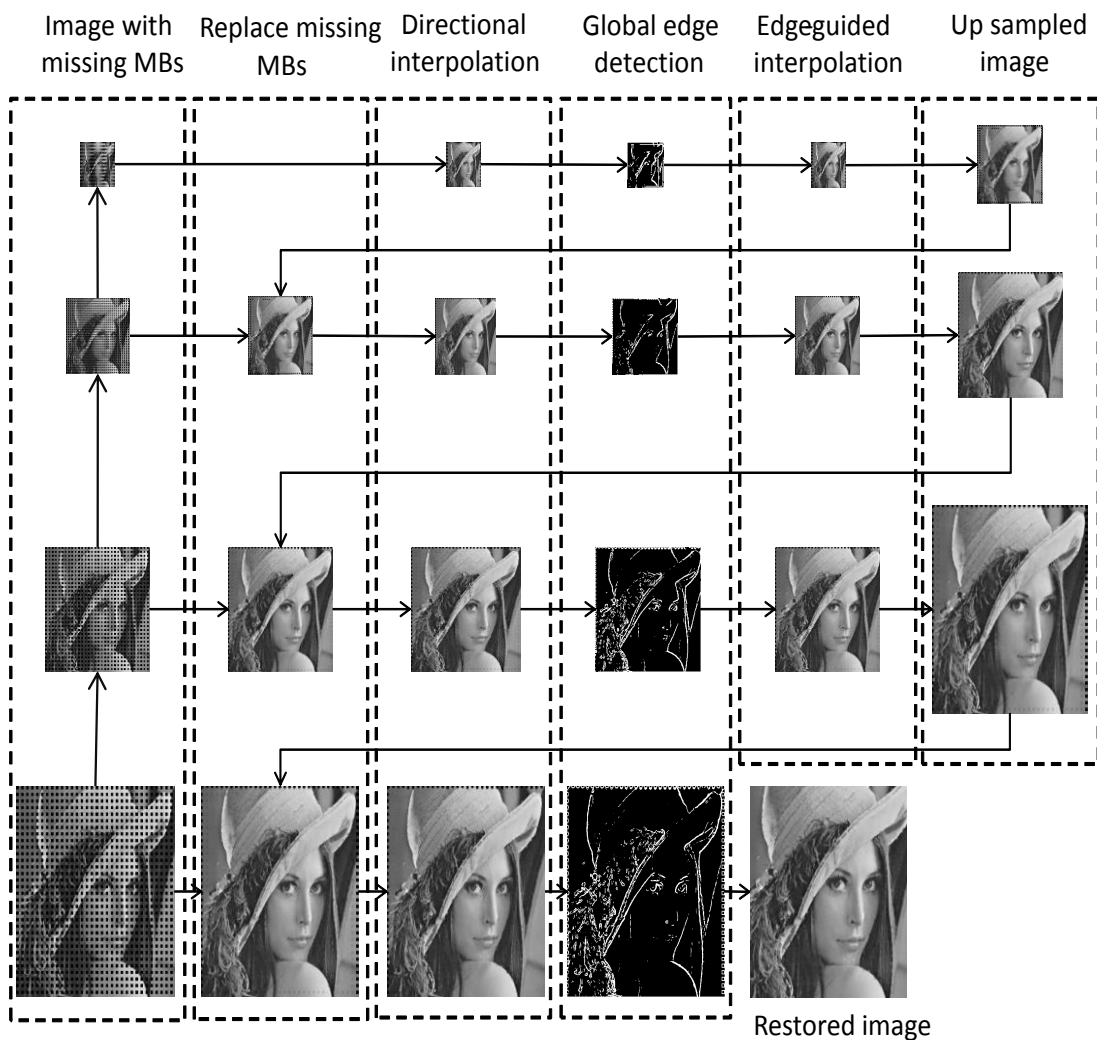


Figure 5.13: Application of the proposed multi-scale method to restoration of a corrupted image of Lena with 25% of 8×8 blocks loss with the Sobel filter.

After edge-based segmentation, the interpolation Equation 5.1 will have its regions of interests (RI), for estimation of the edges, $H, V, C \in RI$. Equation 5.12 would be modified to include edge-segmented (ES) interpolation regions in RI as:

$$A_{m,n} = \sum_{H,V,C \in ES} \sum_{k,l} w_{m+k,n+l} (A_{m+k,n+l} + edge_{m+k,n+l}) \quad (5.12)$$

where now, the interpolation and estimation of the textures and shades in the horizontal, vertical and cross, $H, V, C \in ES$, are confined to edge-segmented (ES) regions composed of relatively homogenous textures. Interpolation using pixels across edge-segmented regions is not allowed at this stage.

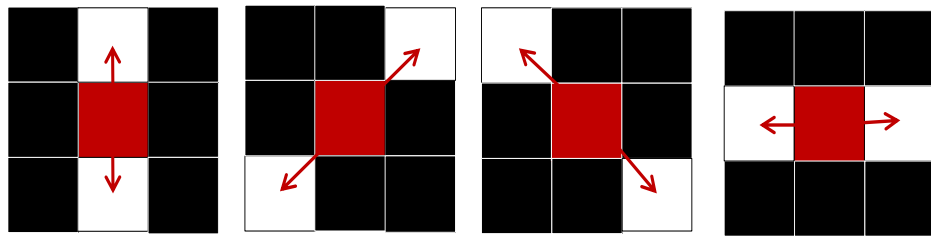


Figure 5.14: Missing pixel (red pixel) might lie in 4 different possible edge directions (white).

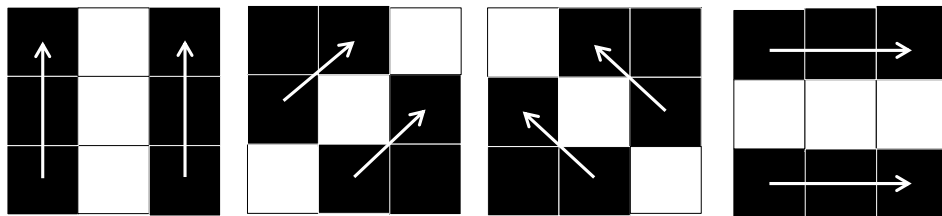


Figure 5.15: Areas next to the edges (white), the missing pixels could be any of the black pixels.

There are two distinct pixel areas that need to be covered separately, with the first being those pixels exactly positioned on the edge boundaries and the second, relates to those that lie next to the edge of the boundaries. Figure 5.14 demonstrates that the missing pixels lie in four different possible edge directions. In this case, in order to estimate these missing pixels it is necessary to first find the direction of the edges, then estimate the missing pixels from those that are on the edge. For the areas that are not entirely on the edge (Figure 5.15,16), but next to the boundaries, the intensity values of pixels on the same side of those surrounding the missing pixels, but not

crossing the edge boundary, should be used, because utilising pixels either on the edge or on the other side of the edge can cause blurred estimation.

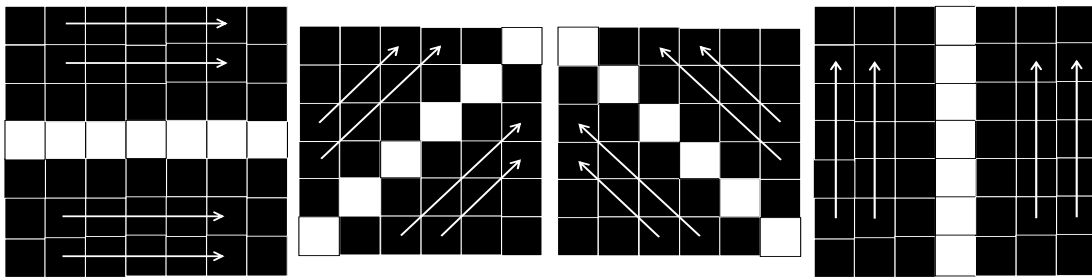


Figure 5.16: Areas next to the edges (white), in the horizontal, left cross, right cross and vertical directions.

Different distances to the boundaries are used at each level of the process, such that longer distances are involved for the higher levels. Figure 5.15 illustrates immediate neighbouring pixels to the edge, and Figure 5.16 shows the second and third levels from the edge areas, where for example, in the last stage fifteen pixels are checked for estimation after each edge.

5.3.1.1 Sobel Edge Detector

The Sobel filter as described in section 3.6.1 is a difference function applied to each pixel in the horizontal $x - axis$ and the vertical $y - axis$ directions. Through rotation the 3×3 mask is able to detect horizontal edges (with a gradient angle of approximately 90°), diagonal edges (with angles of $+45^\circ$ and -45°), and vertical edges (with an angle of 0°).

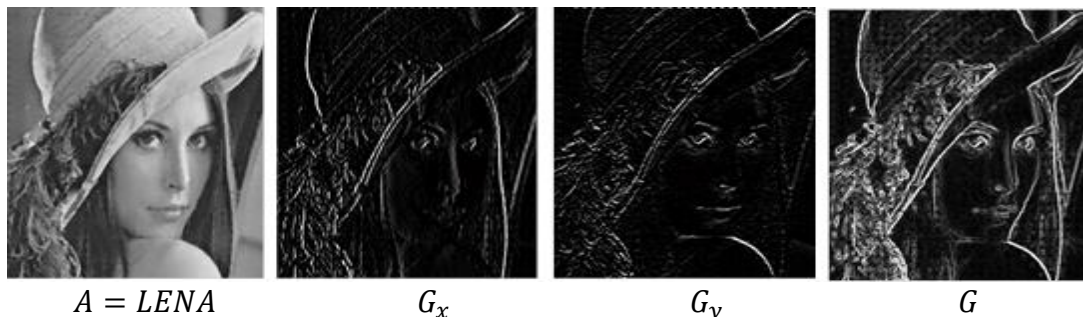


Figure 5.17: Application of the Sobel filter to Lena in x , y and the combined gradient directions.

Figure 5.17 illustrates the application of the Sobel filter to the Lena image and note that in G_x the horizontal edges are enhanced, while in G_y the vertical edges are

enhanced. G , the gradient magnitude of G_x and G_y , displays edges enhanced in both the horizontal and vertical dimensions.

a) Percentile-Rank Threshold Edge-Enhanced Image

Figure 5.17 shows that there are many weak edges due to texture in the gradient image G . In order to retain the most prominent edges only and to suppress the remainder, an edge threshold value, θ_{thresh} , is derived from the α -percentile statistics using the following algorithm:

```

G_sorted = sort(G, 'descend');
index = round( $\alpha * \text{length}(G\_sorted)$ );
 $\theta_{thresh}$  = G_sorted (index);
 $G_{thresh}$  = sign(G,  $\theta_{thresh}$ )

```

Where through a process of experimentation, the fraction α , is set to a value of 0.2 (i.e. 20 percentile statistic) and the function $\text{sign}(G, \theta_{thresh})$ sets the values of $G(i, j)$ less than θ_{thresh} to zero. The application of the threshold results in a differentially processed Lena, as shown in Figure 5.18.

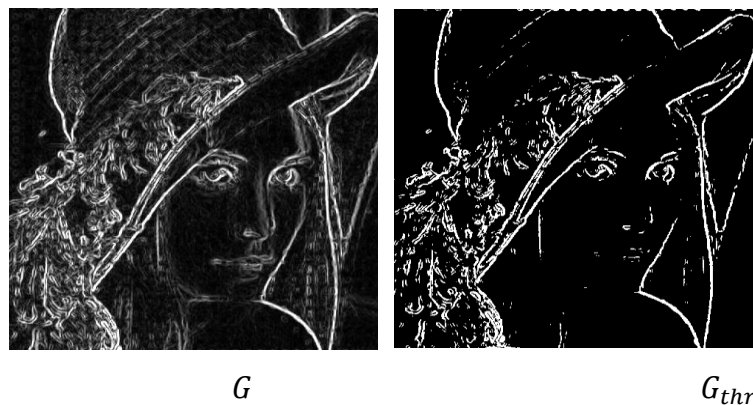


Figure 5.18: Thresholding the output of the Sobel filter, showing input G and output G_{thresh} .

5.3.1.2 Canny Edge Detector

The Canny detector (described in subsection 3.6.2) is a multi-stage algorithm for detection and tracing of the edges in images.

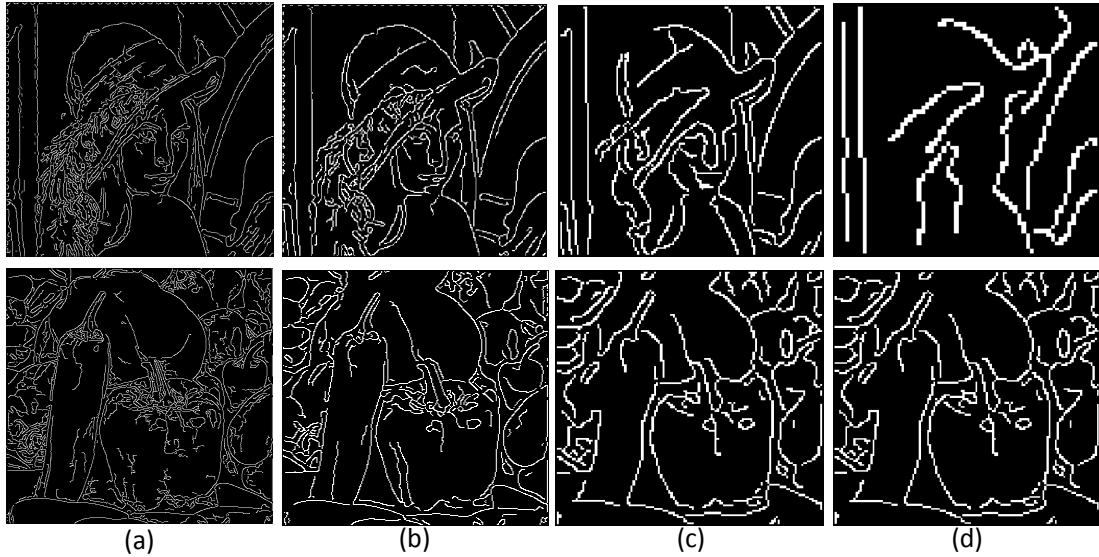


Figure 5.19: Canny edge detector for multi-scale Lena and Peppers at scales, from left to right: (a) 512^2 , (b) 256^2 , (c) 128^2 , (d) 64^2 .

Figure 5.19 shows the application of a Canny detector to multi-scale Lena and Peppers with the image scale progressively down-sampled by 2:1, in three stages, from size 512×512 to 64×64 .

a) Incorporating Global Edge-guided Interpolation in an Iterative Method/Loop

The variance of the Gaussian filter and the maximum and minimum thresholds of the significant edges can be varied to change the sensitivity of the Canny detector. To achieve a better result, different amounts of maximum and minimum thresholds are utilised during the last stage of the process. Three separate iterations (Figure 5.20) run in discrete steps and each has a recovered image as an output and then the restored image is an input for the next iteration. By using repetition the quality of the restored image is improved, as illustrated in Figure 5.20.

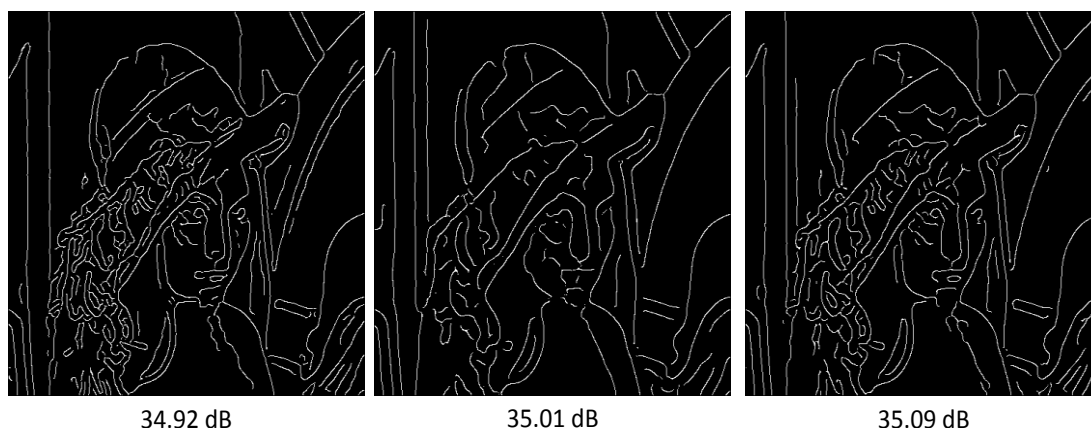


Figure 5.20: Three stage of iteration for Lena and their PSNRs.

5.3.2 Experimental Results

For the performance evaluation results, the proposed algorithm has been tested with the same data as in section 5.2.3. The sizes of the missing blocks are set to 8×8 and 16×16 pixels, with four types of these being evaluated: regular missing macro block at 25% loss rate (Figure 5.21), regular missing macro block at 16×16 (Figure 5.22), random missing macro block with the loss rate set to 10% (Figure 5.23) and random missing macro block with the loss rate set to 40% (Figure 5.24). The choice of the percentage loss is guided by the desire to compare the results with those reported in the literature.

The performance measure criteria used for assessment of the quality of image recovery is the Peak-Signal-to-Noise-Ratio (PSNR), as in the previous sections. Where the domain over which the RMSE is calculated could include only the missing samples or it alternatively, might include the entire image samples composed of the missing and the available samples.

5.3.2.1 Evaluation for 8×8 Regular Loss Pattern

The proposed method is applied to commonly used test images of Lena, Peppers, Man, Boat, Elaine and two further images, including one of the researcher's, added specifically for this work. The PSNR results are compared to a set of twenty one published works representing a number of methods that employ Bayesian and/or edge information for the recovery of regular lost macro blocks. The results are

displayed in two different tables. Table 5.VI represents comparison with the published results, where the PSNR are averaged over the whole image including the available samples and Table 5.VII gives comparison of these results where the PSNR are averaged only over the missing pixels.

To compare the subjective qualities, the simulation results on a missing block size of 8×8 pixels in the Lena, Peppers, Man and Foreman images are shown in Figure 5.21. Observation shows that the proposed method has achieved noticeable improvement in the area of the complex texture structure. In addition, it can be seen that the restored image quality is improved overall when compared with the local edge-guided interpolation technique, especially around the edges, such as in the areas pointed to by red arrows.

Table 5.VI illustrates the performance of the various methods (all the results are directly obtained from Park *et al.* (2005) and Liu *et al.* (2015)). It can be seen that the proposed method performs better than the alternatives considered and there is an improvement in the Canny case of 0.85 dB when compared with the best average performance. Moreover, there is an improvement in the Sobel case over all methods, being 0.79 dB more than the best average performance by Kim, Koo & Jeong (2006) when the PSNRs are computed for the whole image. The methods for comparison of average performance on five images in the Canny case are as follows.

Table 5.VI: Performance comparisons for MB loss rate 25%, MB size= 8×8 , PSNR calculated over whole image: Proposed 1, with the Sobel filter and Proposed 2, with the Canny filter.

Methods	PSNR (dB)					
	<i>Lena</i>	<i>Man</i>	<i>Peppers</i>	<i>Boat</i>	<i>Elaine</i>	<i>Average</i>
Ancis & Giusto (1999)	28.68	25.47	27.92	26.33	29.84	27.65
Sun & Kwok (1995)	29.99	27.25	29.97	27.36	30.95	29.10
Shirani, Kossentini & Ward (2000)	31.69	27.44	31.72	29.22	32.10	30.43

Methods	PSNR (dB)					
	<i>Lena</i>	<i>Man</i>	<i>Peppers</i>	<i>Boat</i>	<i>Elaine</i>	<i>Average</i>
Hemami & Meng (1995)	31.86	27.65	31.83	29.36	32.07	30.55
Alkachouh & Bellanger (2000)	31.57	27.94	32.76	30.11	31.92	30.86
Park <i>et al.</i> (2005)	34.65	29.87	34.20	30.78	34.63	32.83
Kim, Koo & Jeong (2006)	34.91	30.62	35.18	31.40	35.63	33.55
DCT Sobel	35.19	30.56	35.98	31.31	35.43	33.69
DCT Canny	36.08	31.59	36.23	31.76	36.08	34.34

Ancis & Giusto's (1999) algorithm applies the average and average-median to interpolate the missing areas by using the neighbouring blocks edge information and the proposed method's outcome is over by a significant 6.69 dB. Global edge-guided interpolation achieves a PSNR improvement against the POCS-based recovery by Sun & Kwok (1995) of 5.24 dB. Shirani, Kossentini, & Ward (2000) employ inter-block correlation interpolation by using eight weights and linear interpolation to achieve a better result in on diagonal-edge restoration, but this method is also down by 3.91 dB when compared to the proposed method. Hemami & Meng's (1995) technique involves finding four weights rather than eight and has a similar result to Shirani's algorithm, yielding 3.79 dB less than with the proposed method. The DCT transformation is used by Alkachouh & Bellanger (2000) to restore the missing block after DCT, with the high frequency coefficients being set to zero and then the inverse DCT carries out the interpolation, but still the performance of the proposed method is 3.48 dB higher. Park *et al.* (2005) developed an algorithm using the method of alternating projection, which is based on orthogonal projections onto constraint sets in Hilbert space. It can achieve better outcomes when compared with the previous methods, but the global edge-guided interpolation performance is above this by 1.51 dB. The last method in this section is that of Kim, Koo & Jeong (2006), which employs fine directional interpolation by using a spatial directional vector and

achieves the best result out of all of the extant techniques, but this is still lower than the global edge-guided interpolation method by 0.79 dB.

Additionally, the global edge-guided interpolation also improved the result of local edge-guided interpolation and moreover, it can be seen that the Canny filter performs better than the Sobel filter, as it expected from the literature.

Table 5.VII demonstrates the PSNR values taken from Li & Orchard (2002), Zhao *et al.* (2005) and Zhai *et al.* (2010) directly and calculated on the region of the missing sample blocks only. The outcomes show that the proposed DCT Canny achieves a higher result compared with the DCT Sobel method by 1.72 dB. The methods for comparison of average performance on thirteen published works in the Canny case are as follows.

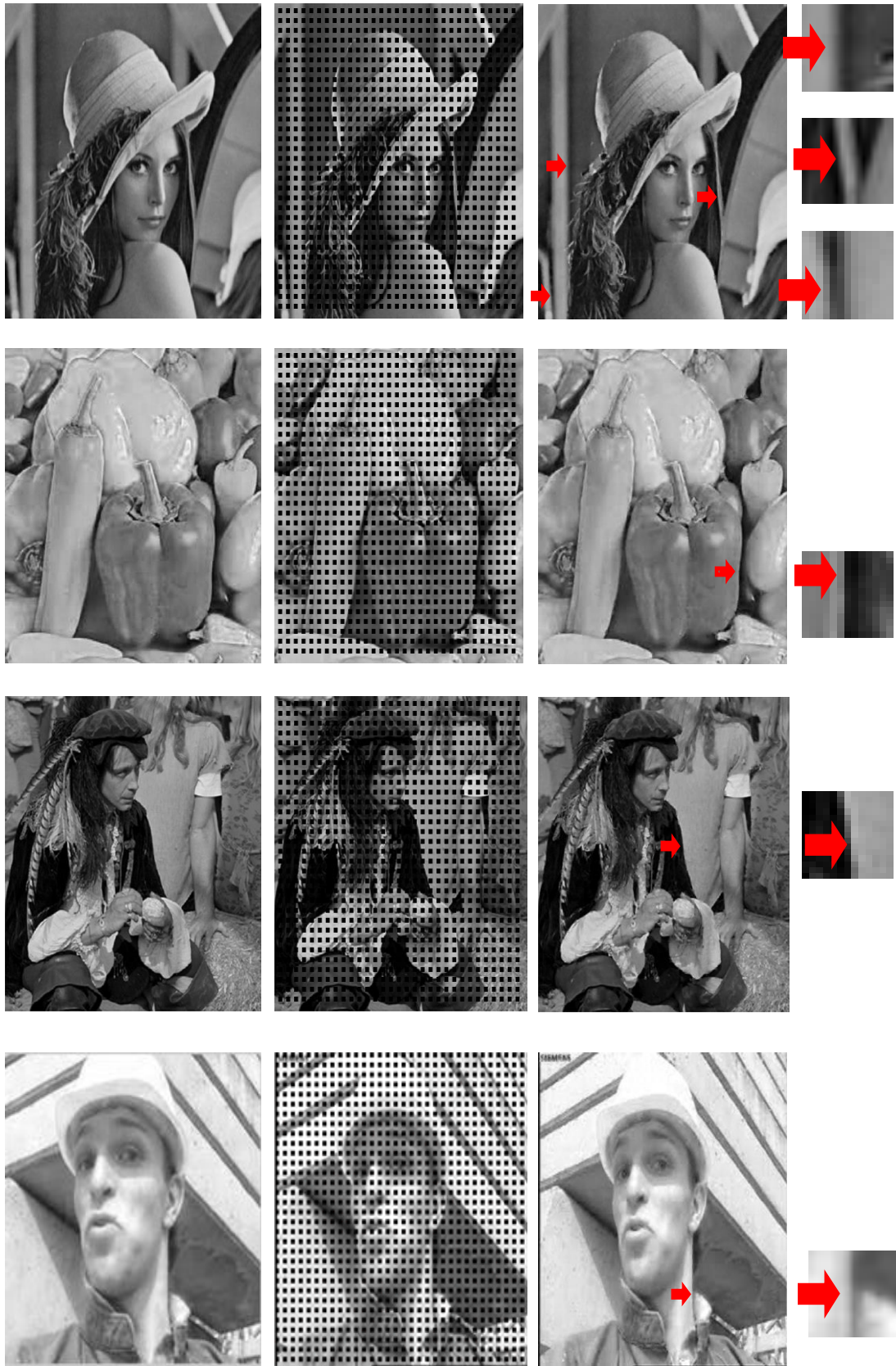


Figure 5.21: From left to right, the original images, the image with 25% missing MBs and the restored images for Lena, Peppers, Man and Foreman.

Agrafiotis, Bull & Canagarajah's (2006) method represented the worst performance and the current proposed method achieves an improvement of 6.81 dB over theirs. Li & Orchard (2002) introduced sequential error-concealment, such that the previously recovered pixels can be used in the recovery process afterwards, but compared with global edge-guided interpolation the outcome is less by 1.53 dB. Jung, Chang & Lee (1994), Park et al. (2005) and Zhao *et al.* (2005) obtain almost similar results, such that the proposed method is better by 3.44 dB, 3.78 dB and 2.85 dB, respectively. Zeng & Liu (1995) made use of the local geometric information extracted from the surrounding blocks to interpolate the missing pixels and their PSNR outcome was 2.35 dB lower than for the proposed method. The rest of techniques have approximately the same results, ranging from Alkachouh & Bellanger (2000) at 24.00 dB to Hsia (2004) at 25.14 dB, but the global edge-guided interpolation provides a higher outcome when compared to theirs. The best performance among all the compared methods belongs to Zhai *et al.* (2010) at 28.51 dB, who employed a Bayesian and multi-scale estimation approach with DCT and yet, the proposed method can also improve this result by 1.27 dB.

Table 5.VII: Performance comparisons for MB loss rate of 25%, MB size= 8×8, PSNR calculated just for the region of the missing blocks for Lena: Proposed 1, with the Sobel filter and Proposed 2, with the Canny filter.

Methods	Image Lena	Methods	Image Lena
	PSNR (dB)		PSNR (dB)
Zhai <i>et al.</i> (2010)	28.51	Zhao <i>et al.</i> (2005)	26.93
Agrafiotis, Bull & Canagarajah (2006)	22.97	Hsia (2004)	25.14
		Sun & Kwok (1995)	24.70
Park <i>et al.</i> (2005)	26.00	Shirani, Kossentini, & Ward (2000)	24.50
Zhai <i>et al.</i> (2008)	28.11	Zeng & Liu (1995)	27.43
Li & Orchard (2002)	28.25	Alkachouh & Bellanger (2000)	24.00
Jung, Chang & Lee (1994)	26.34	DCT Sobel	28.06
Wang, Zhu & Shaw (1993)	24.70	DCT Canny	29.78

Table 5.VIII demonstrates the PSNRs values taken from Liu *et al.* (2015) directly and were calculated over whole image for Lena. The outcomes show that the proposed DCT Canny achieves a highest result among all the published works.

Table 5.VIII: Performance comparisons for MB loss rate of 25%, MB size= 8×8, PSNR calculated over whole image for Lena with the Canny filter.

Methods	Image Lena	Methods	Image Lena
	PSNR (dB)		PSNR (dB)
Sun & Kwok (1995)	29.99	Park <i>et al.</i> (2005)	34.65
Ancis & Giusto (1999)	28.68	Zhai <i>et al.</i> (2008)	34.45
Hemami & Meng (1995)	31.86	Rongfu, Yuanhua & Xiaodong (2004)	34.07
Shirani, Kossentini & Ward (2000)	31.69	Kim, Koo & Jeong (2006)	34.91
Alkachouh & Bellanger (2000)	31.57	Liu <i>et al.</i> (2015)	35.70
Varsa, Hannuksela & Wang (2001)	32.05	Kolada <i>et al.</i> (2015)	33.74
Li & Orchard (2002)	35.70	DCT Canny	36.08

5.3.2.2 Evaluation for 16 × 16 Regular Loss Pattern

A block loss size of 16 × 16 pixels is also applied to the proposed algorithm for further evaluation and a 5-level DCT pyramid is required to implement the proposed method. Figure 5.22 demonstrates the subjective performance of the original, damaged and reconstructed images. In addition, Table 5.IX shows the PSNR comparison between the proposed algorithm and previous works.

The 16 × 16 isolated block loss is a most challenging case since many rows of image blocks are entirely lost. Table 5.IX contains an objective comparison between the global edge guided interpolation method and six published works on the Lena and Peppers images. The outcomes show an improvement of global edge-guided over the local edge-guided interpolation by 2.32 dB and the Sobel edge-guided interpolation is not as effective as the Canny edge-guided interpolation. From the

simulation results, the average for the proposed DCT edge-guided scheme has superior PSNR over all six published works of Salama, Shroff & Delp (1998), by 3.37 dB, Wang, Zhu & Shaw, (1993) by 2.98 dB, Sun & Kwok (1995) by 4.15 dB, Park *et al.* (2005), by 2.49 dB, Li & Orchard (2002) by 0.36 dB and Kim, Koo & Jeong (2006) by 0.07 dB.

Table 5.IX: Performance comparisons for MB loss size = 16×16, PSNR calculated over whole image on Lena and Peppers: with the DCT Sobel and DCT Canny filters.

Methods	PSNR (dB)		
	<i>Lena</i>	<i>Peppers</i>	<i>Average</i>
Salama (Salama, Shroff & Delp, 1998)	35.01	34.71	34.86
Wang (Wang, Zhu & Shaw, 1993)	35.43	35.07	35.25
Sun (Sun & Kwok, 1995)	34.95	33.21	34.08
Park (Park <i>et al.</i> 2005)	35.98	35.50	35.74
Li (Li & Orchard, 2002)	37.48	38.27	37.87
Kim (Kim, Koo & Jeong, 2006)	37.37	38.95	38.16
Sobel Edge-guided	37.48	38.82	38.15
DCT Edge-guided	37.57	38.91	38.23

5.3.2.3 Evaluation for 8 × 8 Random Loss Pattern

The random loss pattern involves missing macro blocks in random positions that may include a random sequence of adjacent horizontal and/or vertical losses. In order to evaluate the performance result for the random block loss, the proposed method is applied to the Lena, Peppers, Man and Foreman images, degraded with random block loss, as shown in Figure 5.23. The positions of the missing macro blocks are random and distinct in each evaluation test, therefore the program is applied in a number of iterations to find the mean PSNR distortion, and the number of iterations is set to a value of ten for this part (Table 5.X), as explained in subsection 5.2.3.2. Then, the results are compared with several published methods, and Table 5.XI includes those published techniques that are taken from Li & Orchard, (2002) and Zhai *et al.* (2010) directly. The PSNRs are only for the missing pixels and hence, do not include the available samples.

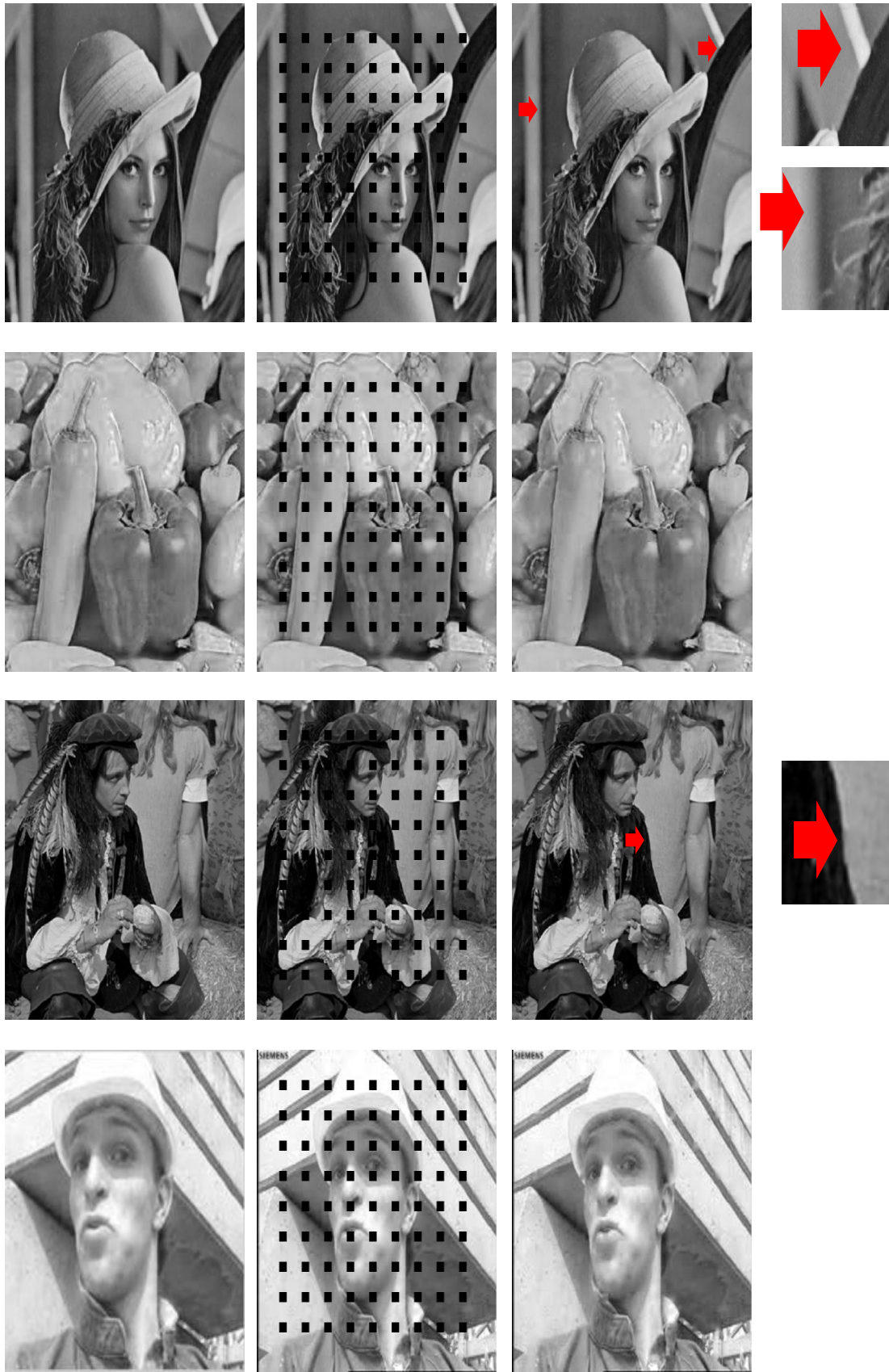


Figure 5.22: From left to right, the original images, corrupted images (16×16 isolated block loss), the restored images and zoomed in images for Lena, Peppers, Man and Foreman (Canny case).

Table 5.X: Performance comparisons for random MB loss rate of 10% (MB size = 8×8) on Lena with ten steps of iteration.

Image Result	Lena									
	1	2	3	4	5	6	7	8	9	10
PSNR (dB)	33.35	32.84	32.20	32.59	32.83	33.14	33.01	32.67	32.52	32.86
AVG.	32.80									

As displayed in Table 5.XI, the proposed method performs better than the rest, and there is an improvement for the Canny detector by 4.67 dB when compared with the best performance among all the prior results. The lowest outcome is that of Zeng & Liu (1995), which is 6.20 dB under, what has been achieved here. Zhai *et al.* (2008) proposed two methods, the first, image error concealment via block-based bilateral filtering and the second, Bayesian error concealment with a DCT Pyramid (Zhai *et al.*, 2010) that does achieve reasonably higher performance at 28.25 dB, but global edge-guided surpasses both of these by 5.15 dB and 4.67 dB, respectively. Finally, Li & Orchard (2002) presented sequential error-concealment, whereby previously recovered pixels are used in the recovery process afterwards and compared to the current method it has a lower PSNR by 5.42 dB. It can be seen that the Sobel filter cannot provide competitive results for the proposed method appearing to be inferior to nearly all the other methods.

Table 5.XI: Performance comparisons for a random MB loss rate of 10%, MB size = 8×8, PSNR calculated just for the region of missing MBs for Lena: Proposed 1, with the Sobel filter and Proposed 2, with the Canny filter.

Methods	Image Lena
	PSNR (dB)
Zhai <i>et al.</i> (2010)	28.13
Zhai <i>et al.</i> (2008)	27.65
Zeng & Liu (1995)	26.60
Li & Orchard (2002)	27.38
DCT Sobel	30.76
DCT Canny	32.80



Figure 5.23: From left to right; the original Lena, Peppers, Man and Foreman images, the image with 10% random missing MBs and the restored image.

The proposed method was tested with a high missing macro blocks rate of 40%, size (8×8) . Figure 5.24 shows the performance on Peppers, and the outcome has clearly improved the user experience, compared with the corrupted image.



Figure 5.24: From left to right; the original Peppers image, the image with 40% random missing MBs and the restored image.



Figure 5.25: Top: from left to right, the original Lena and the image with 25% regular missing MBs (8×8) . Bottom: from left to right, the reconstructed image with local edge-guided interpolation and global edge-guided interpolation.

Figure 5.25 clearly shows that the global edge-guided interpolation achieves better results, especially around the main edges, when compared with local edge-guided interpolation as well as the various published methods, which have been reported on in the previous sections.



Figure 5.26: From left to right, the original image, corrupted image (8×8 isolated block loss) and the restored image for my own picture, PSNR at 36.60 dB.

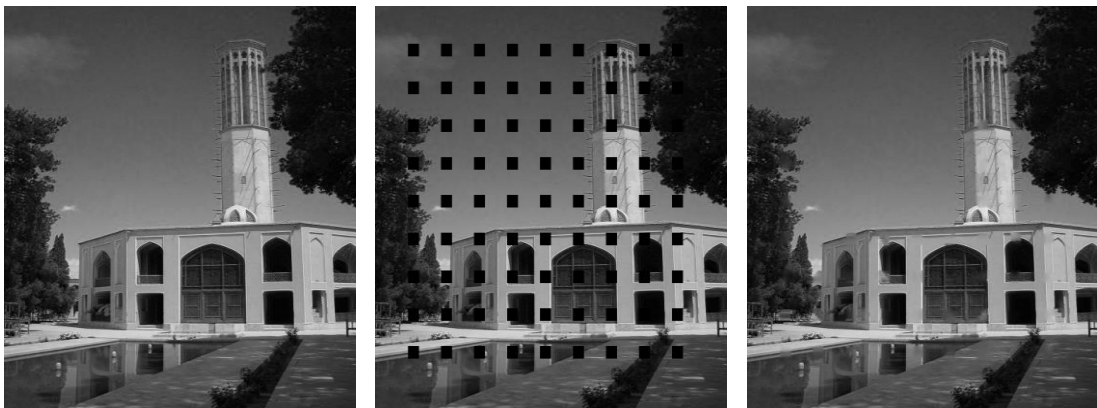


Figure 5.27: From left to right, the original image, corrupted image (16×16 isolated block loss) and the restored image for Dolat-Abad Garden, Yazd, Iran.

In addition the proposed algorithm has been tested on some miscellaneous images, which have been taken by the researcher (Figure 5.26 and Figure 5.27).

Figure 5.28 illustrates the performance comparison between the proposed method and six published works on the Man image. The outcome shows that the proposed method improved the result by 1.69 dB compared with the best performance among all the above mentioned methods.

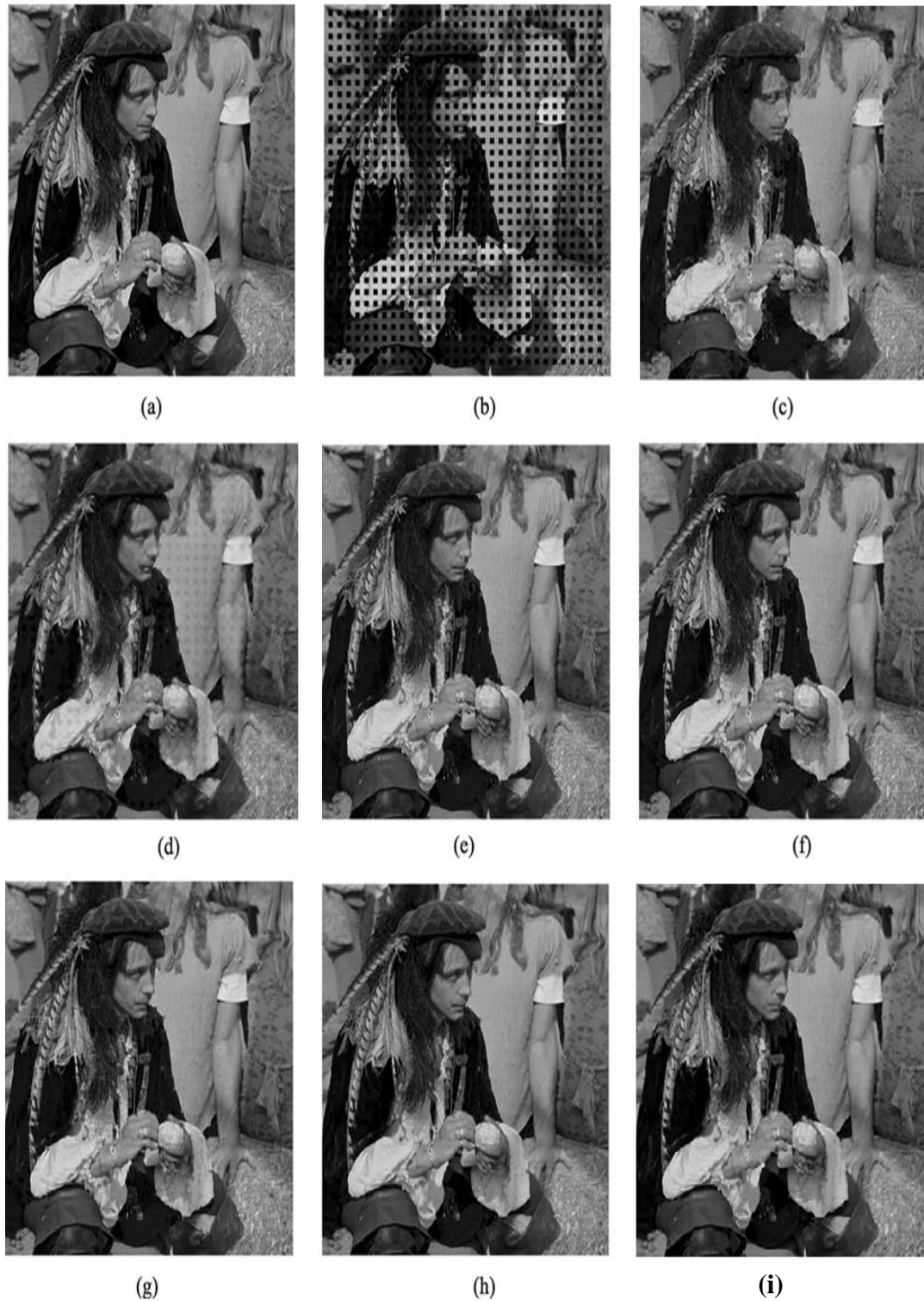


Figure 5.28: Experiment on a lost block size of 8×8 pixels of the “Man” image. (a) Original image 512×512 , (b) damaged image of one missing block out of every four. Restoration using the methods of (c) Ancis and Giusto (PSNR = 25:47 dB), (d) Sun and Kwok (PSNR = 27:25 dB), (e) Hemami and Meng (PSNR = 27:65 dB), (f) Shirani *et al.* (PSNR = 27:44 dB), (g) Alkachouh and Bellanger (PSNR = 27:94 dB), (h) Park *et al.* (PSNR = 29:87 dB), (i) proposed method (31.56 dB).

5.4 Conclusion

In this chapter, an image gap restoration method with application to packet loss concealment in networks, where image MB may be lost due to congestion or signal fading has been evaluated. The main contributions of the work are the inclusion of local and global edge enhancement strategies within a pyramid DCT image processing framework. The proposed algorithm includes a combination of multi-resolution transforms, directional interpolation and edge-guided enhancement capable of restoring missing MBs including the edges. The experimental results demonstrate that significant improvement in the quality and PSNR of the restored images are obtained by the proposed edge-guided image restoration method.

Moreover, this chapter presents improvements in the image gap restoration of chapter four through incorporation of edge-based directional interpolation within a multi-scale pyramid transform. Two types of image edges have been reconstructed: (a) the local edges or textures inferred from the gradients of the neighbouring pixels and (b) the global edges or boundaries between image objects or segments, inferred using different types of edge detector applications. Through a process of pyramid transformation and down-sampling, the image has been transformed into a series of progressively reduced size layers until at the pyramid apex the gap size is one sample. The process is then reversed and at each stage, the missing samples are estimated, up-sampled and combined with the available samples.

The DCT pyramid has been used for comparison with published works and evaluations over a range of images have demonstrated the proposed method provides a better PSNR. In addition, it has been concluded that the Canny edge detector improve the interpolation compared with the Sobel edge detector.

It has been observed from the performances of various different methods of interpolation for an MB loss rate of 25% and MB size of 8×8 on well-known test images, where the PSNR was calculated for the whole image, that the DCT Canny edge detector offers the best average result (at 34.40 dB). The same outcomes are obtained in the case of random an MB loss rate of 10% and a regular MB loss rate of size 16×16 , being 32.80 dB and 38.23 dB respectively. Moreover, the results

surpass those achieved in Chapter 4. Further work in the next chapter involves the use of complex wavelets instead of DCT for pyramid decomposition.

Chapter 6

Multi-scale Edge-Guided Wavelet Image Gap Restoration

Chapter 6

6. Multi-scale Edge-Guided Wavelet Image Gap Restoration

6.1 Introduction

Wavelet theory naturally lends itself to pyramid decomposition and reconstruction. In this chapter, a novel multi-scale pyramid method using wavelet transform is proposed as an alternative to DCT-pyramid image gap reconstruction, as described in Chapter 5. The wavelet pyramid incorporates, as an alternative to a DCT-pyramid, conventional and edge-guided interpolation. The results are compared with those of DCT-pyramid interpolation in relation their capacities for filling the corrupted regions in damaged images.

Through a process of wavelet pyramid transformation, the image is transformed into a series of progressively reduced size layers in two separate parts, approximation or low-pass components and details or high-pass components, until at the pyramid apex, where the gap size is reduced to one sample. The process is then reversed; at each stage, the missing samples are inferred using estimates, which may be a combination of the local, global edges and also restored details information that is then up-sampled and combined with the available uncorrupted samples.

There are two main benefits of this combination. First, the pyramid method starts interpolation from the smaller damaged macro block size, which makes it easier to replace the missing parts. Second, wavelet transformation keeps the processes of high and low pass information separate (Padmavathi , Priyalakshmi & Soman, 2012).

In order to replace missing regions of the image with the help of surrounding pixel information that is already present in the original image, the process for wavelet based edge guided interpolation is in the main the same as the DCT based edge-guided interpolation technique. The same methods of interpolation are applied to the approximation section of the damaged image and in addition to that by using the wavelet transform, the values of three details parts (vertical, horizontal and diagonal) are ascertained. Then, these details parts are restored in the missing parts by using the available uncorrupted information. Finally, the two results are combined in order to achieve the result, and it is found that the outcome of the process is improved and more efficient compared with previous published works. There are two possible alternative approaches to the wavelet pyramid:

- (1) Wavelet decomposition and re-composition of the whole distorted images;
- (2) Wavelet decomposition and re-composition of macro block segmented images.

These will be considered in detail next.

6.2 Wavelet Analysis-Synthesis of Distorted Images

Wavelet transformation is one of the main techniques for time-frequency signal transformations. Discrete wavelet transform (DWT) is used in this method as it can analyse the signal with various resolutions at different frequency bands by a simple procedure, as it decompose/separates the original signal into two parts, approximation and details, thus allowing independent analysis of the coefficients at different scales (Maxim & Zakharova, 2012).

6.2.1 Components of Wavelet Decomposition; Approximation and Details

An example of image decomposition is shown in Figure 6.1 for Lena (512×512) with a block loss size of 8×8, and is decomposed into four quadrants with different interpretations (HH, HL, LH and LL).













Size Part	256×256	128×128	64×64
cA Approx.			
cH Detail			
cV Detail			
cD Detail			

Figure 6.1: Components of wavelet decomposition; approximation and details for lossy Lena (block loss size = 8×8), in 256×256, 128×128 and 64×64.

The successive application of two-dimensional DWT leads to a decomposition of approximation coefficients at level j in four components: the approximation at level $j + 1$, and the details in three orientations (horizontal, vertical, and diagonal) that are related mostly to image edges. It can be seen in Figure 6.1, top, that the two-dimensional wavelet decomposition computes the approximation coefficients matrix cA and details coefficients matrices cH , cV , and cD (horizontal, vertical, and diagonal, respectively), obtained by wavelet decomposition of the input image.

6.2.2 Types of Wavelength Applicable to the Gap Restoration Problem

The characteristics of the signal/image and also the nature of the application influence the selection of the type of wavelet. Consequently, a good understanding of the attributes of the analysis and wavelet can lead to the choice of one that is optimal for each application. As discussed in chapter 3, there are a large number of wavelets that can be used for both discrete and continuous analysis. Morlet, Meyer, derivative of a Gaussian and Paul wavelets are examples of the continuous, whereas orthogonal form, such as Daubechies and B-spline bi-orthogonal wavelets are discrete forms that are suitable for the proposed method. One of the important features that aids image restoration is multi-resolution analysis. However, it needs to be noted that all of the discrete wavelets cannot offer multi-resolution analysis and a Journe wavelet is one of these.

Two methods of wavelet processing will be investigated, one for the whole distorted image and the second for macro block segmented images.

6.2.2.1 Wavelet Analysis-Synthesis over the Whole Distorted Images

Table 6.I illustrates the performance of nine different types of wavelets and their applications for the restoration of Lena with 25% missing macro blocks and Figure 6.2 shows the visual results of restoration for each wavelet form. It can be seen from both Table 6.I and Figure 6.2 that the best performances among all the compared methods are $db1$ and $bior1.1$ at 35.72 dB, thus these can be used as wavelet transforms in the whole distorted images interpolation method. The next five

methods dmey, sym2, bior2.2, rbio5.5 and db2 have PSNRs of 10.83 dB, 9.32 dB, 8.8 dB, 5.06 dB and 7.88 dB lower than the best methods, respectively. Coif5 has the worst outcome (22.59 dB) compared with db1, performing 13.13 dB below it.

Figure 6.2 shows some of the performance results for certain wavelet types. In order to investigate this performance, the wfilters ('wname') command is used in Matlab to make comparisons for all nine methods in Table 6.I. Generally, wfilters ('wname') compute two pairs of filters, first low-pass, high-pass decomposition filters (LoD,HiD), and then second, low-pass, high-pass reconstruction filters (LoR,HiR), which are associated with the orthogonal or bi-orthogonal wavelet, named in the string 'wname'. Thus, each filter includes its own returns coefficient approximations.

Table 6.II shows the number and the values of the coefficients for each filter within wavelet type. It can be seen from Table 6.I and Table 6.II that the best results for the proposed algorithm are the ones that have a short length of only two values. During pyramid decomposition, longer length wavelets tend to smear (mix) the boundaries of the distorted regions into those of the undistorted ones at successive levels of decomposition and hence, cause degradation of the overall results.

Table 6.I: Performance comparisons using different wavelet types for MB loss size (8×8), PSNR over the whole image for Lena, with a missing data rate of 25%.

Wavelets					
Lena	coif5	Dmey	sym2	bior1.1	bior2.2
PSNR (dB)	22.59	24.89	26.40	36.12	26.96
SSIM	0.7386	0.7974	0.8154	0.9666	0.8377
Wavelets					
Lena	rbio1.1	rbio5.5	db1	db2	
PSNR (dB)	36.12	30.66	36.12	27.84	
SSIM	0.9666	0.9269	0.9668	0.8425	

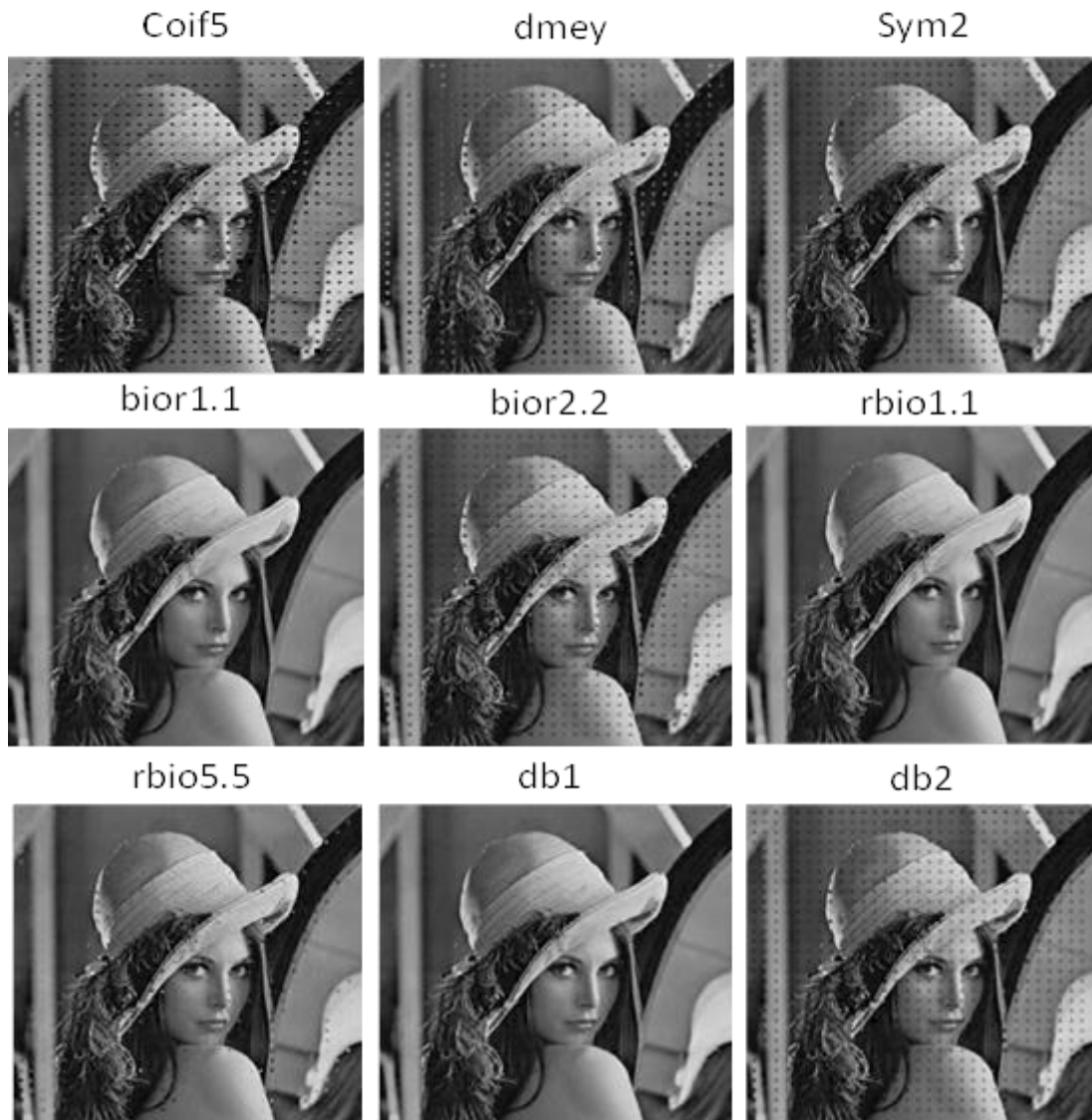
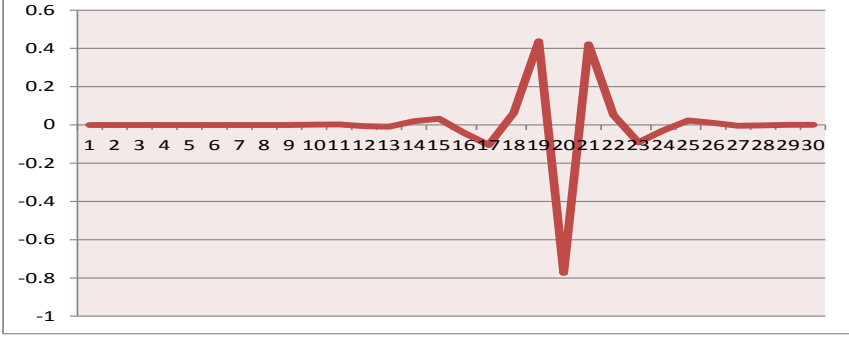
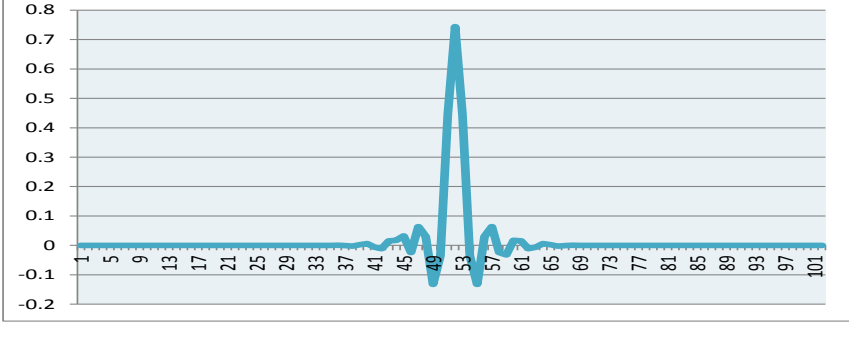
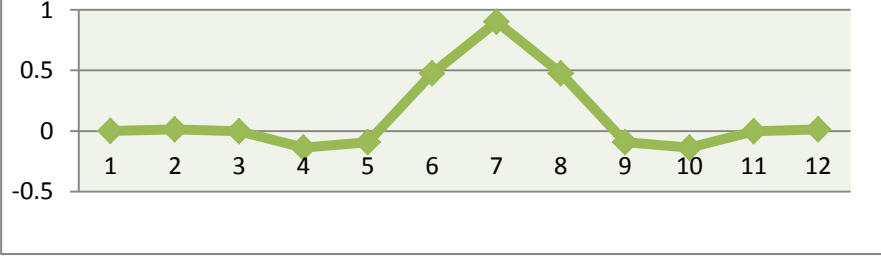


Figure 6.2: Performance comparisons using nine different wavelet types for the whole distorted Lena (macro block loss size = 8×8).

6.2.2.2 Wavelet Analysis-Synthesis over Macro-Blocks of Distorted Images

To overcome the degradation of the overall results when using the higher length wavelets, which have presented in Table 6.I and Table 6.II, macro block segmentation is applied to the image. Regarding this method, the wavelet is applied to each macro block separately. As a result, distorted regions are not combined with undistorted ones at successive levels of decomposition and hence, the overall outcomes are improved. The results for this are shown in Table 6.III.

Table 6.II: Performance comparisons using nine different wavelet types, showing the number of coefficients in the decomposition low-pass filter (Lo_D) part on Lena for a macro block loss size = 8×8.(y-axis : coefficients number , x-axis : values).

Wavelets Type	Wavelet Coefficient Sequence
coif5	 <p>The graph for 'coif5' shows a red line representing the coefficient sequence. The x-axis is labeled from 1 to 30, and the y-axis ranges from -1 to 0.6. The sequence starts near 0, has a small peak around coefficient 15, a sharp negative peak around coefficient 19, and another sharp positive peak around coefficient 21.</p>
Dmey	 <p>The graph for 'Dmey' shows a blue line representing the coefficient sequence. The x-axis is labeled from 1 to 101, and the y-axis ranges from -0.2 to 0.8. The sequence is mostly flat near 0, with a prominent positive peak around coefficient 53.</p>
sym2	-0.1294 0.2241 0.8365 0.4830
bior1.1	0.7071 0.7071
bior2.2	0 -0.1768 0.3536 1.0607 0.3536 -0.1768
rbio1.1	0.7071 0.7071
rbio5.5	 <p>The graph for 'rbio5.5' shows a green line with diamond markers representing the coefficient sequence. The x-axis is labeled from 1 to 12, and the y-axis ranges from -0.5 to 1. The sequence starts at 0, has a small negative dip at coefficient 4, a peak at coefficient 7, and returns to 0 by coefficient 12.</p>
db1	0.7071 0.7071
db2	-0.1294 0.2241 0.8365 0.4830

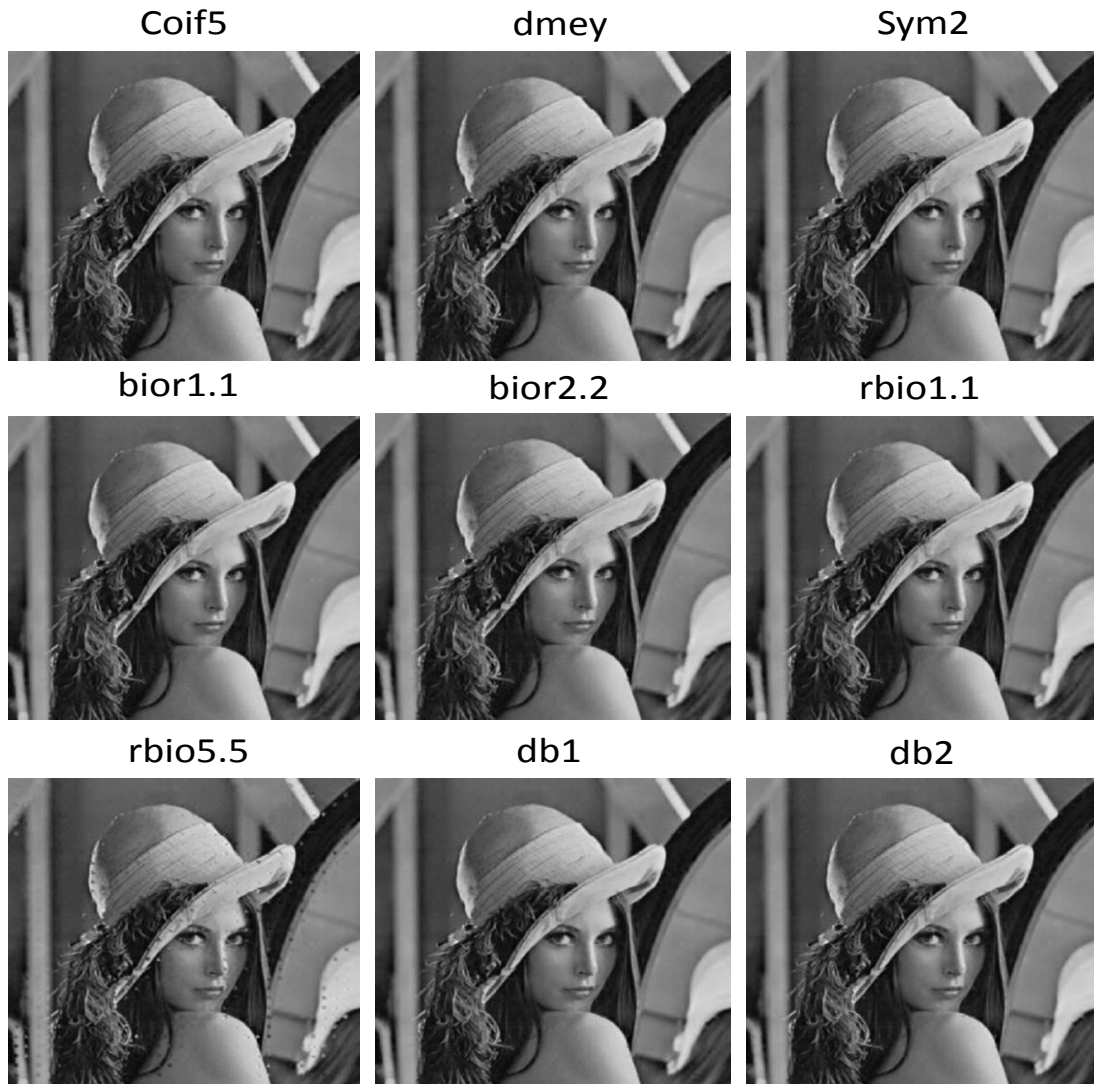


Figure 6.3: Performance comparisons using nine different wavelet types for the macro block segmented distorted Lena (macro block loss size = 8×8), distortion pattern = regular, distortion rate = 25%.

Nine different types of wavelets and their applications to the restoration of macro-block segmented Lena with 25% missing macro blocks is shown in Table 6.III. The following six methods achieved higher results compared with outcomes of the whole distorted image, namely, coif5, dmey, sym2, bior2.2, rbio5.5 and db2. Their PSNRs are 11.83 dB, 10.55 dB, 9.43 dB, 8.71 dB, 3.44 dB and 7.99 dB higher than the previous results, respectively. Figure 6.3 shows the visual results of the restoration for each wavelet form.

It can be seen from both Figure 6.3 and Tables 6.III, that although the performances of higher length wavelets are improved with the use of macro-block segments, the best results still belong to db1 and bior1.1 at 36.12 dB. Thus, in this chapter, the db1

wavelet is used as the transform in both interpolation approaches over the whole distorted images and over the macro block segmented images, in order to achieve the best outcome.

Table 6.III: Performance comparisons using different wavelet types for MB loss 25% (MB size= 8×8), PSNR calculated over the macro-block segmented image on Lena.

Wavelets					
Lena	coif5	Dmey	sym2	bior1.1	bior2.2
PSNR (dB)	34.42	35.44	35.83	36.12	35.67
SSIM	0.9620	0.9683	0.9691	0.9702	0.9688
Wavelets					
Lena	rbio1.1	rbio5.5	db1	db2	
PSNR (dB)	36.12	34.10	36.12	35.83	
SSIM	0.9702	0.9652	0.9702	0.9691	

6.2.2.3 Daubechies db1 (Haar) wavelet

Since the Daubechies db1 wavelet is that of choice for this work, it is explained in some further detail. This wavelet, also known as the Haar wavelet, is the only orthogonal wavelet with a linear phase. Linear phase refers to it having a filter with all of its frequency components having proportional phase change that are shifted over time by the same constant amount. The mother wavelet function ψ and the scaling function ϕ of db1 are illustrated in Equations 6.1 and 6.2:

$$\psi(t) = \begin{cases} 1 & \text{if } 0 \leq t < 1/2, \\ -1 & \text{if } 1/2 \leq t < 1, \\ 0 & \text{otherwise.} \end{cases} \quad (6.1)$$

$$\phi(t) = \begin{cases} 1 & \text{if } 0 \leq t < 1, \\ 0 & \text{if otherwise.} \end{cases} \quad (6.2)$$

Figure 6.4 demonstrates the scaling function and wavelet function for the db1 wavelet.

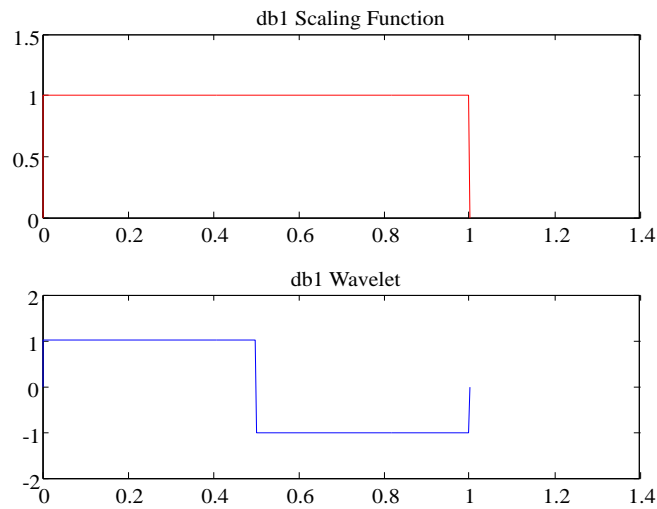


Figure 6.4: The scaling function and wavelet for Daubechies' wavelet (db1).

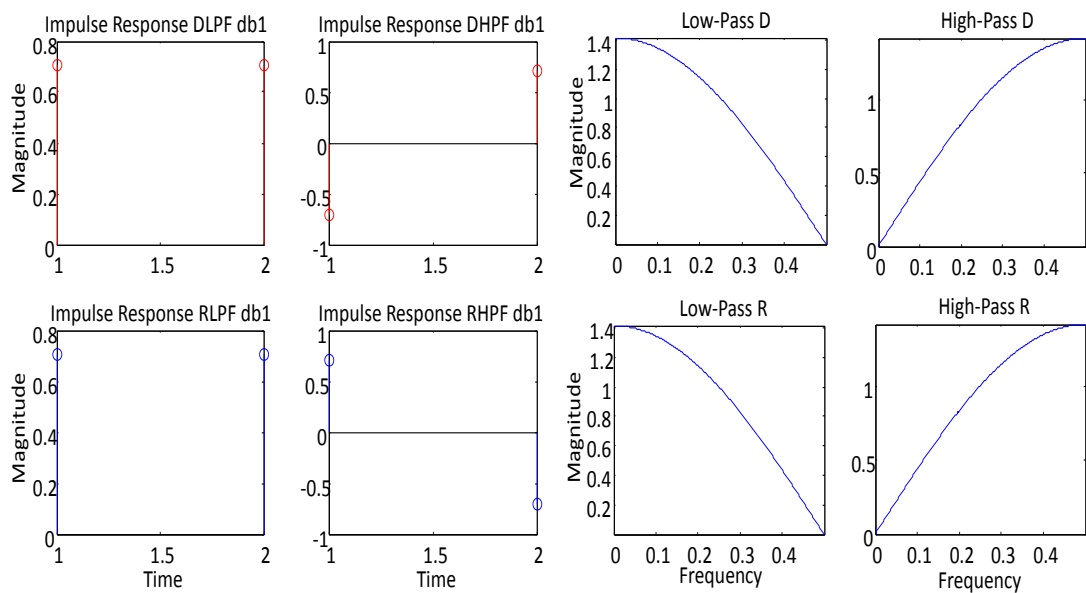


Figure 6.5: Impulse responses (left) and frequency responses (right) of the decomposition and reconstruction filters for the db1 bi-orthogonal wavelet.

The impulse response and the frequency response of the decomposition and reconstruction filters are plotted in Figure 6.5. As shown, each filter stage is composed of a low-pass filter and its quadrature mirror image high pass filter.

6.3 Wavelet Reconstruction of Distorted Images – Conventional vs Edge-Guided Interpolation

6.3.1 Conventional (non-edge-based) Interpolation

Tables 6.IV and 6.V show the results when two simple conventional methods, namely, median and mean, are solely utilised as estimators for missing pixels at the apex of a wavelet coefficient pyramid. While the mean estimate does not include any edge information the median estimate has a measure of edge preservation built in. Although the median estimator achieves the higher result in comparison with the mean method (as it includes edge information in the estimation of the missing pixel), it is not still sufficient and hence, a more advanced process is necessary to improve the outcome.

Table 6.IV: Performance comparisons for an MB loss rate of 25% (MB size = 8×8) on Lena, Man, Peppers, Boat and Elaine with a mean estimator at the apex.

Performance measure	Images					
	<i>Lena</i>	<i>Man</i>	<i>Peppers</i>	<i>Boat</i>	<i>Elaine</i>	<i>Average</i>
PSNR (dB)	29.05	27.62	29.99	27.52	30.29	28.89

Table 6.V: Performance comparisons for an MB loss rate of 25% (MB size = 8×8) on Lena, Man, Peppers, Boat and Elaine with a median estimator at the apex.

Performance measure	Images					
	<i>Lena</i>	<i>Man</i>	<i>Peppers</i>	<i>Boat</i>	<i>Elaine</i>	<i>Average</i>
PSNR (dB)	29.47	27.72	30.46	27.79	30.40	29.16

6.3.2 Edge-Guided Interpolation

Figure 6.6 illustrates the method in the case when *local* edge guided information is added to the previous technique. As observed in Figure 6.7 and also in section 5.4,

preserving the *local* edges mitigates blurring distortions of textures and provides improved interpolation at the *local* texture level, in particular, at the boundaries of the available and missing samples.

Thus, local edge guided interpolation is used in the approximation part in two steps. First, it is applied to the apex of the multi-scale pyramid with the size of the missing block being reduced to one, as explained in 5.2.1. Second, after interpolation of the apex sample, local edge guided interpolation is utilised for the subsequent stages for blocks of size $N \times N$. Starting from the outer boundaries of the macro block, the missing pixels are progressively replaced towards the centre by utilising the surrounding available pixels edges in the horizontal, vertical and cross directions, the equation regarding this is explained in 5.2.2.

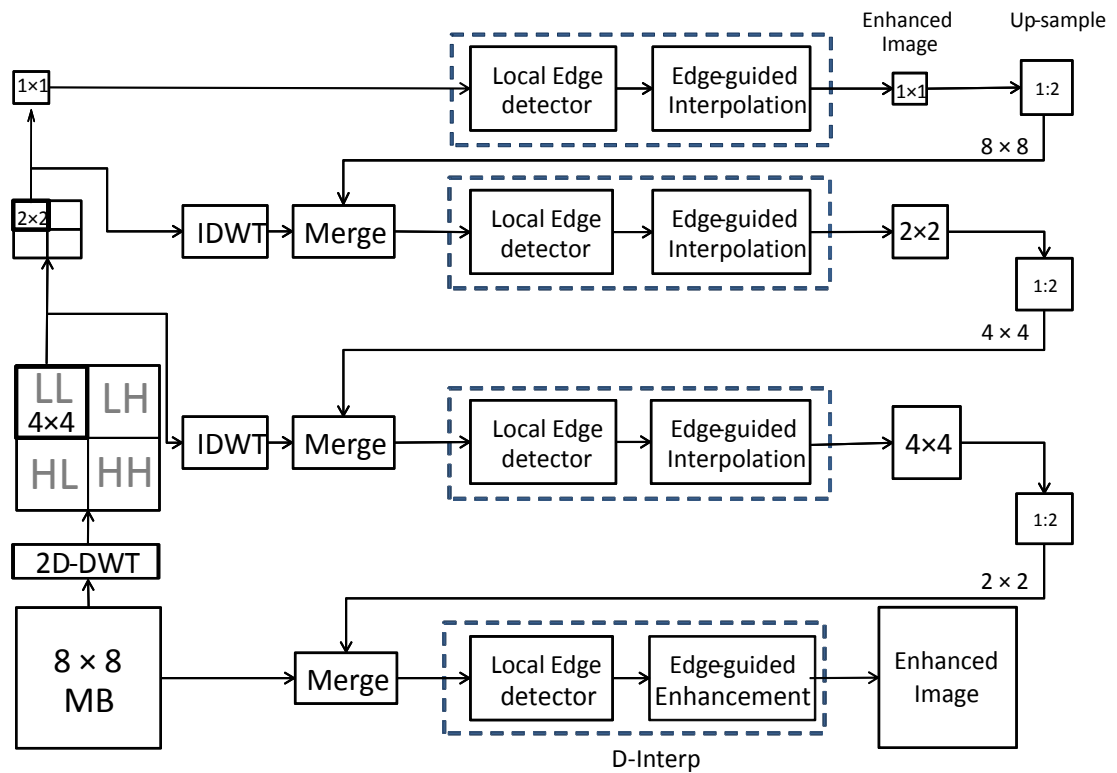


Figure 6.6: Diagram of the process of local-edge guided wavelet interpolation.

Figure 6.7 shows that using the *local* edge information is not sufficient to obtain a satisfactory result, especially around the edges, such as in the areas pointed at by red arrows.

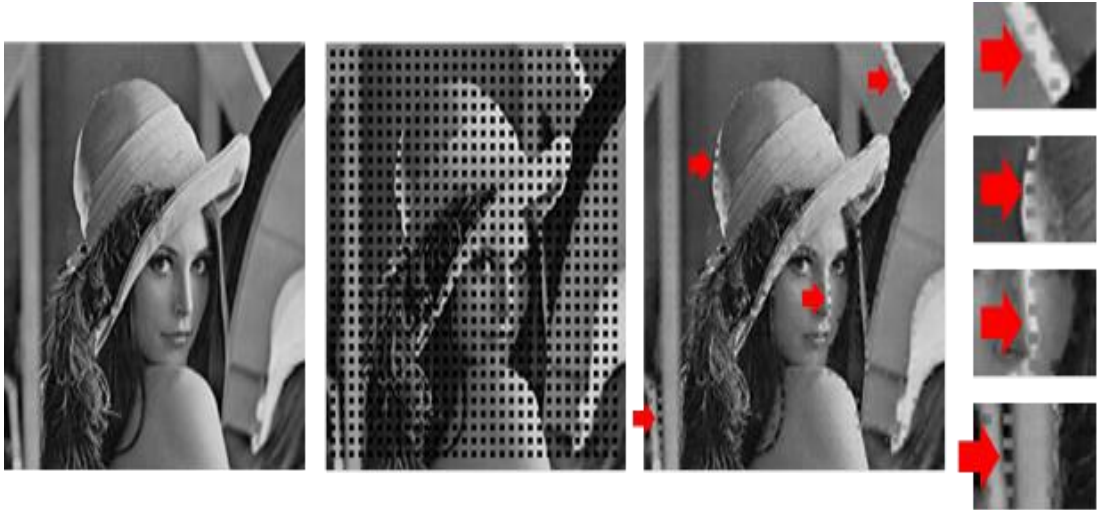


Figure 6.7: From left to right, the original image, the image with 25% MB loss (8×8 isolated block loss) and the restored image for Lena with local-edge guided wavelet interpolation.

For further improvement where the missing blocks contain significant edges, such as major boundaries between segments and objects, the *global* edge information, not necessarily evident within the lost macro blocks, needs to be utilised separately for all four coefficient sections (approximation and three details parts). The process is as follows:

- 1) Decompose the image into a wavelet pyramid structure and the result includes four sections: one approximation and three details coefficient parts. The apex of the pyramid represents the last stage of down-sampling (where each macro block of size 8×8 is reduced to only one pixel after three stages, and for an macro bloock of size 16×16 is reduced to only one pixel after four stages);
- 2) Starting from the apex of the wavelet pyramid in the approximation part, interpolate the decimated (down-sampled) gap using the local edge information from the neighbouring pixel, as described in section 5.2;
- 3) Using a Canny edge detector, track the global edges in the interpolated images;
- 4) Enhance the interpolated gap estimation using the global edge information;
- 5) Starting from the apex of the wavelet pyramid in the three details parts, separately interpolate the decimated (down-sampled) gap using the local edge information from the neighbouring pixel;

- 6) Combine the two outcomes from stages four and five;
- 7) Up-sample the enhanced interpolated image via a zero-padded inverse 2D-wavelet and combine/merge with the available received samples of the same layer of up-sampling;
- 8) Go to step (1) and repeat the process for each intermediate stage of up-sampling;
- 9) In the last stage of up-sampling the number of iterations in step (3) and step (4) is used to improve the result further.

The details of these sub-processes are illustrated in Figure 6.8.

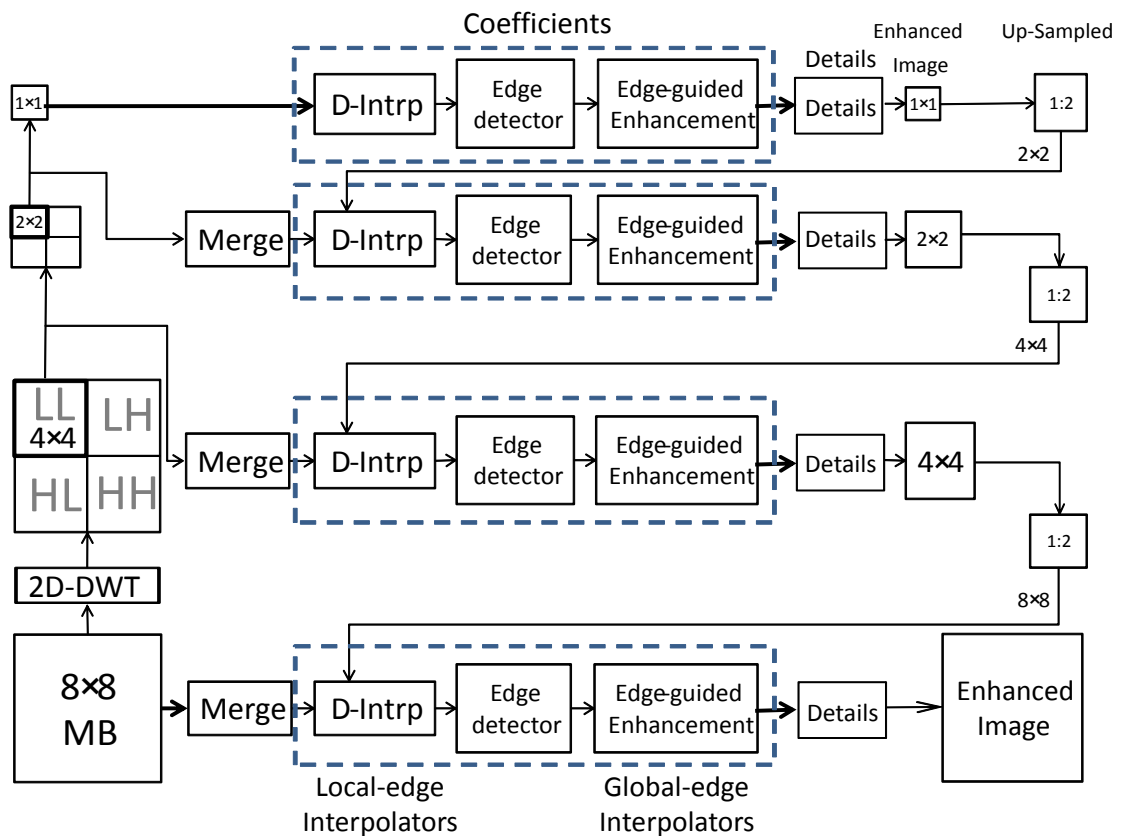


Figure 6.8: Diagram showing the process of local, global edge-guided interpolation for the coefficients parts, which is followed by the addition of the restored details part (D-Intrp = Directional Interpolation).

A substantial improvement is shown in Figure 6.9 compared to Figure 6.7 after using the global edge guided interpolation; the main edges are recovered and the overall image enhancement quality is improved.

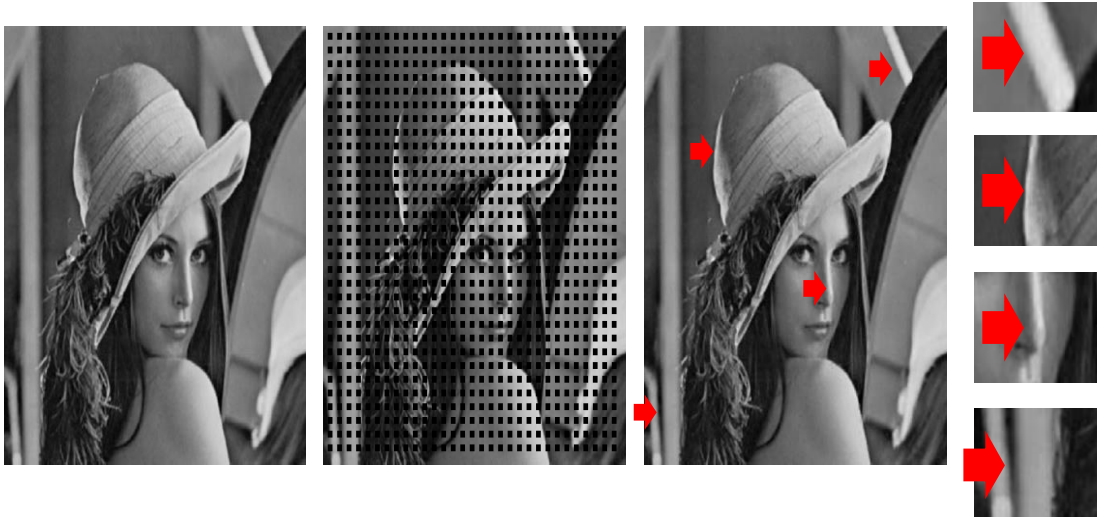


Figure 6.9: From left to right, the original Lena, Lena with 25% macro block loss (8×8) and restored Lena with global edge guided interpolation.

6.3.2.1 Wavelets and Edge Information

The main cause of substantial interpolation errors and visible distortions in image restoration comes from the wrong interpolation across the significant edges. To avoid or mitigate the problem of blurred/smeared interpolation, global edge interpolation is applied to the approximation and details parts, singly.

First: the Canny edge detector is applied to the coefficient part (cA), and an edge-based segmentation of an image into two broadly homogenous-texture areas is provided. Figure 6.10 shows the application of a Canny detector to a multi-scale wavelet Lena with the image scale progressively down-sampled by 2:1, in three stages, from size 512×512 to 64×64 . Then, the outcome can be used as a guide to enhance the result from the previous section based on equations presented in 5.3.1.



Figure 6.10: Canny edge detector for a multi-scale wavelet on Lena at scales, from left to right: (a) 512^2 ; (b) 256^2 ; (c) 128^2 ; and (d) 64^2 .

Second: details in three orientations, cH , cV , and cD (horizontal, vertical, and diagonal), which relate mostly to image edges, are utilised. Figure 6.11 illustrates a block diagram of the three-stage wavelet pyramid image decomposition and its application to the Lena image for a missing block size of 8×8 . It shows that the details coefficients cH , cV , and cD provide advantageous edge information.

As details coefficients matrices cH , cV , and cD are affected by missing pixels, a restoration process needs to be applied to this information before using it. Figure 6.12 clearly demonstrates improvement in the details coefficients matrices cH , cV , and cD after performing this restoration process. Subsequently, the image information can be combined with the rest of the repaired available data, namely cA .

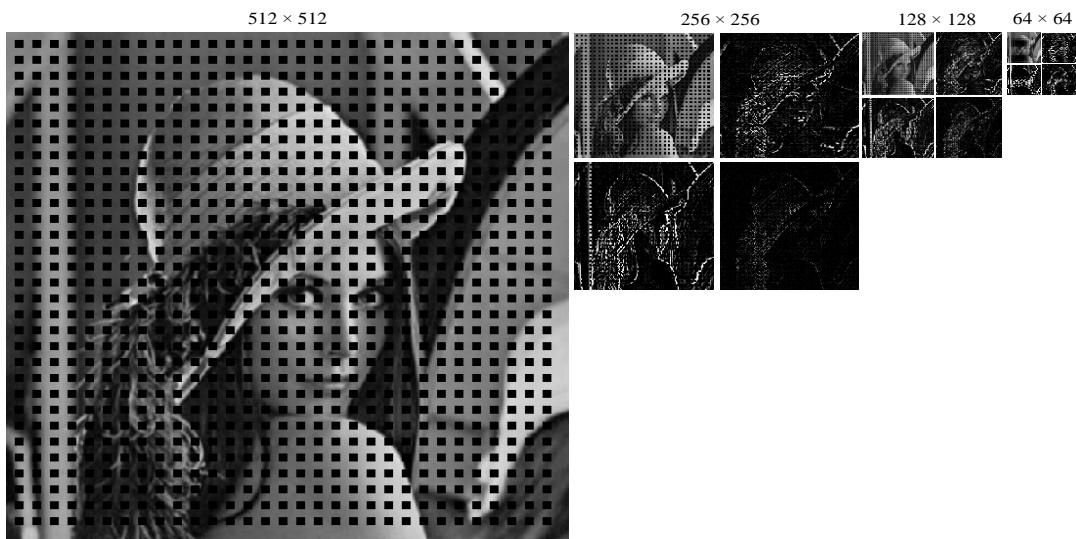


Figure 6.11: Block diagram of the three-stage wavelet pyramid image decomposition on the Lena image for missing block size of 8×8 .

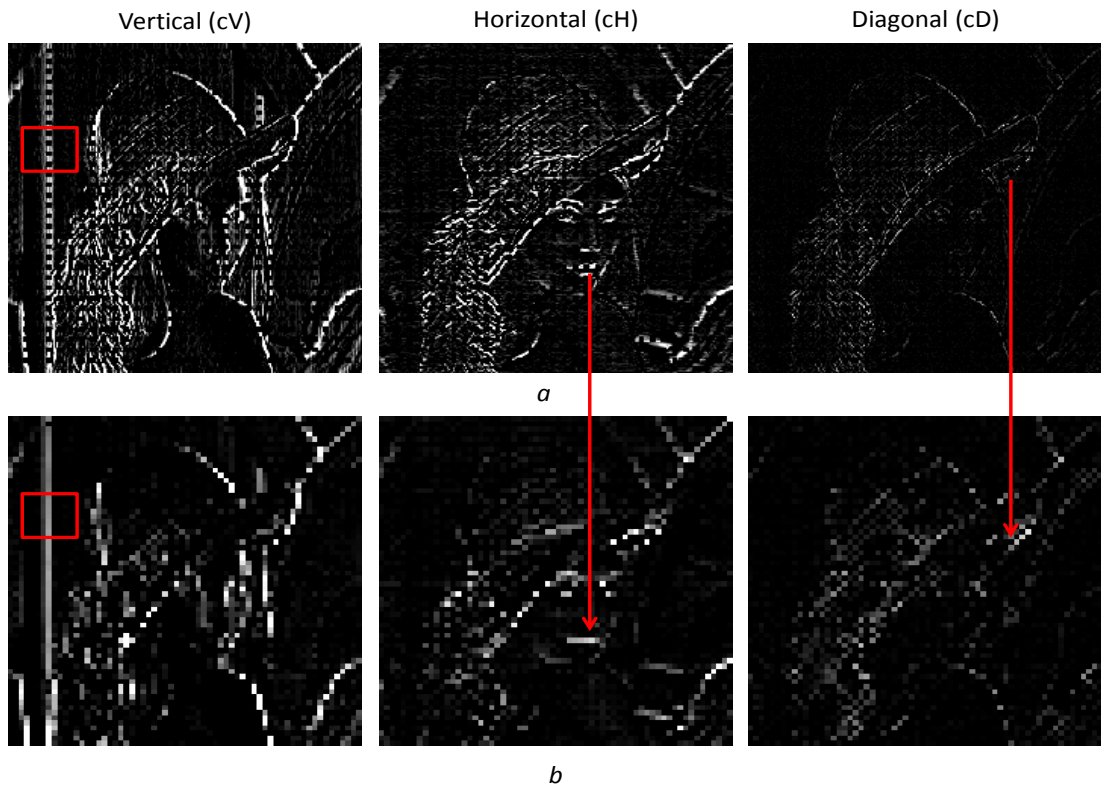


Figure 6.12: (a) affected coefficients matrices, cV, cH, and cD by missing pixels (8×8); and (b) the restored cV, cH and cD on Lena.

a) Iterative Estimation of Edges and Approximation Coefficients

The iterative method produces a gain in PSNR. It starts the process by using a high level of Canny edge details at the first iteration, and then continues the process by reducing the amount of these details as a guide for the interpolation.

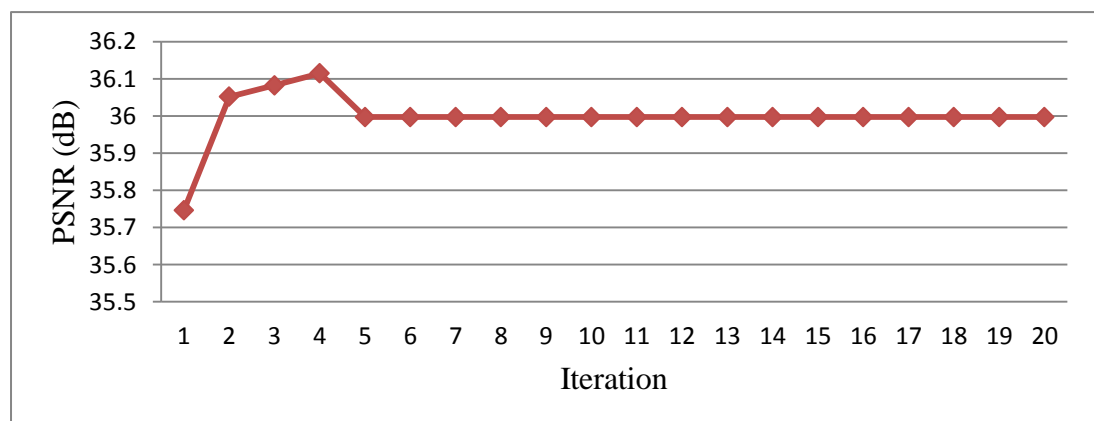


Figure 6.13: Performance comparisons with twenty iterations for a macro block loss rate of 25% (8×8) on Lena for the last stage of reconstruction of the cA coefficient part.

Figure 6.13 shows the PSNR values obtained from twenty steps of iteration on Lena for a macro block loss rate of 25% (8×8), and the increase from the first to the fourth stage is from 35.74 dB to 36.12 dB. Subsequently, it is observed that for further iterations the amount of PSNR stays constant and hence, the number is set to four for this method. Figure 6.14 demonstrates four stages of iteration on Lena.



Figure 6.14: Four stages of iteration for Lena starting from the left (first stage) and finishing on the right (fourth stage).

6.3.2.2 Threshold Details Coefficients-Enhanced Image

It can be seen in Figure 6.12 that there is some unnecessary information in all the details coefficients. Those redundant data might cause false recovery in the restored image and therefore, it is essential to remove them. The designed method for this tries to keep wavelet coefficients that include good image information and eliminate the rest. Hard-thresholding is used in the proposed algorithm by checking each pixel intensity in all three details parts and it sets any coefficient less than or equal to the threshold of zero (Equation 6.3). Figure 6.15 shows graphs of both the original and discarded wavelet coefficients after applying hard-thresholding.

$$coif(i) = \begin{cases} coif(i) & \text{if } coif(i) > \theta_{thresh} \\ 0 & \text{if } coif(i) \leq \theta_{thresh} \end{cases} \quad (6.3)$$

where, $coif(i)$, represents each pixel in the coefficients matrices cH, cV and cD.

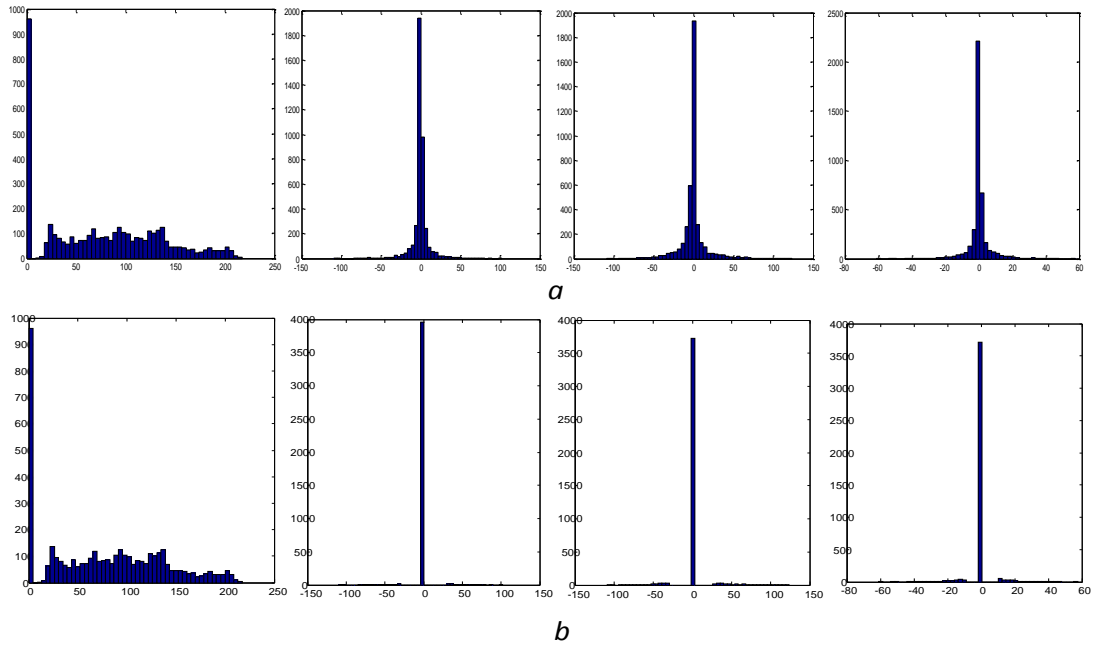


Figure 6.15: (a) original graphs of cA , cH , cV and cD ; and (b) graphs of cA , cH , cV and cD after thresholding.



Figure 6.16: (a) original Lena; (b) Lena with 25% macro block loss (8×8); (c) Lena with macro block loss (16×16); and (d) Lena with 10% random MB loss (8×8).

6.4 Evaluation for Regular Loss Pattern

For performance evaluation results, the proposed algorithm has been tested on the same data as in section 5.2.3. The sizes of the missing blocks were set to 8×8 and 16×16 pixels and five types of these were evaluated: regular missing macro block at 25% loss rate, regular missing macro block at 16×16 , random missing macro block with the loss rate set to 10% (Figure 6.16) and also random missing macro block with the loss rate set to 25% and 40% (Figure 6.20). The choice of the percentage loss was guided by the desire to compare the results with those reported in the literature. The performance measurement criteria used for assessment of the quality of image recovery are same as in the previous chapters: Peak-Signal-to-Noise-Ratio (PSNR).

6.4.1 Regular Loss Pattern 8×8 Missing Block

In this subsection, experimental results are reported that show the performance of regular loss pattern with a missing block size of 8×8 and with the number of inner iterations set to four for the cA coefficient part at the last stage of up-sampling. There are two ways of calculating PSNR employed: the case when it is solely computed over the whole image and that where it is calculated just for the missing parts region.

Figure 6.17 demonstrates the original, erroneous and reconstructed images after applying the proposed error concealment method and it can be seen that the result is not blurry, with the shape having been recovered correctly. The results of this experiment are given in Table 6.VI when the PSNR is calculated over the whole image and it is observed that the propose method produces the best PSNR at 34.31 dB compared to seven previously published methods. In addition, the global edge-guided interpolation has improved upon the result of the local edge-guided interpolation.

Table 6.VI: Performance comparisons for MB loss rate 25%, MB size=8×8, PSNR calculated over the whole image, with the canny filter.

Methods	PSNR (dB)					
	<i>Lena</i>	<i>Man</i>	<i>Peppers</i>	<i>Boat</i>	<i>Elaine</i>	<i>Average</i>
Ancis & Giusto (1999)	28.68	25.47	27.92	26.33	29.84	27.65
Sun & Kwok (1995)	29.99	27.25	29.97	27.36	30.95	29.10
Shirani, Kossentini & Ward (2000)	31.69	27.44	31.72	29.22	32.10	30.43
Hemami & Meng (1995)	31.86	27.65	31.83	29.36	32.07	30.55
Alkachouh & Bellanger (2000)	31.57	27.94	32.76	30.11	31.92	30.86
Park <i>et al.</i> (2005)	34.65	29.87	34.20	30.78	34.63	32.83
Kim, Koo & Jeong (2006)	34.91	30.62	35.18	31.40	35.63	33.55
Wavelet Canny Edge-guided	36.12	31.61	36.35	31.79	36.13	34.40

Ancis & Giusto’s (1999) algorithm applies the average and average-median to interpolate the missing areas by using the neighbouring blocks edge information and the proposed method’s outcome is over by a significant 6.75 dB. Proposed wavelet edge-guided interpolation achieves a PSNR improvement against the POCS-based recovery by Sun & Kwok (1995) of 5.30 dB. Shirani, Kossentini, & Ward (2000) employ inter-block correlation interpolation by using eight weights and linear interpolation to achieve a better result in on diagonal-edge restoration, but this method is also down by 3.97 dB when compared to the proposed method. Hemami & Meng’s (1995) technique involves finding four weights rather than eight and has a similar result to Shirani’s algorithm, yielding 3.85 dB less than with the proposed method. The DCT transformation is used by Alkachouh & Bellanger (2000) to

restore the missing block after DCT, with the high frequency coefficients being set to zero and then the inverse DCT carries out the interpolation, but still the performance of the proposed wavelet method is 3.54 dB higher. Park *et al.* (2005) developed an algorithm using the method of alternating projection, which is based on orthogonal projections onto constraint sets in Hilbert space. It can achieve better outcomes when compared with the previous methods, but the wavelet edge-guided interpolation performance is above this by 1.57 dB. The last method in this section is that of Kim, Koo & Jeong (2006), which employs fine directional interpolation by using a spatial directional vector and achieves the best result out of all of the extant techniques, but this is still lower than the wavelet edge-guided interpolation method by 0.85 dB.

The PSNR is also calculated for the region of missing blocks, for a missing block loss of 8×8 on Lena. Table 6.VII shows the outcomes for twelve published methods with PSNR values taken from Belfiore *et al.* (2003) and Kim, Koo & Jeong (2006) directly and it can be seen that the current proposed method achieves an improvement over all of the others.

Table 6.VII: Performance comparisons for MB loss rate of 25%, MB size= 8×8, PSNR calculated just for the region of missing blocks for Lena, with the Canny filter.

Methods	Image Lena	Methods	Image Lena
	PSNR (dB)		PSNR (dB)
Zhai <i>et al.</i> (2010)	28.51	Zhao <i>et al.</i> (2005)	26.93
Agrafiotis, Bull & Canagarajah (2006)	22.97	Hsia (2004)	25.14
		Sun & Kwok (1995)	24.70
Park <i>et al.</i> (2005)	26.00	Zeng & Liu (1995)	27.43
Zhai <i>et al.</i> (2008)	28.11	Li & Orchard (2002)	28.25
Shirani, Kossentini, & Ward (2000)	24.50	Alkachouh & Bellanger (2000)	24.00
Jung, Chang & Lee (1994)	26.34	Wavelet edge-guided	29.83
Wang, Zhu & Shaw, 1993	24.70		

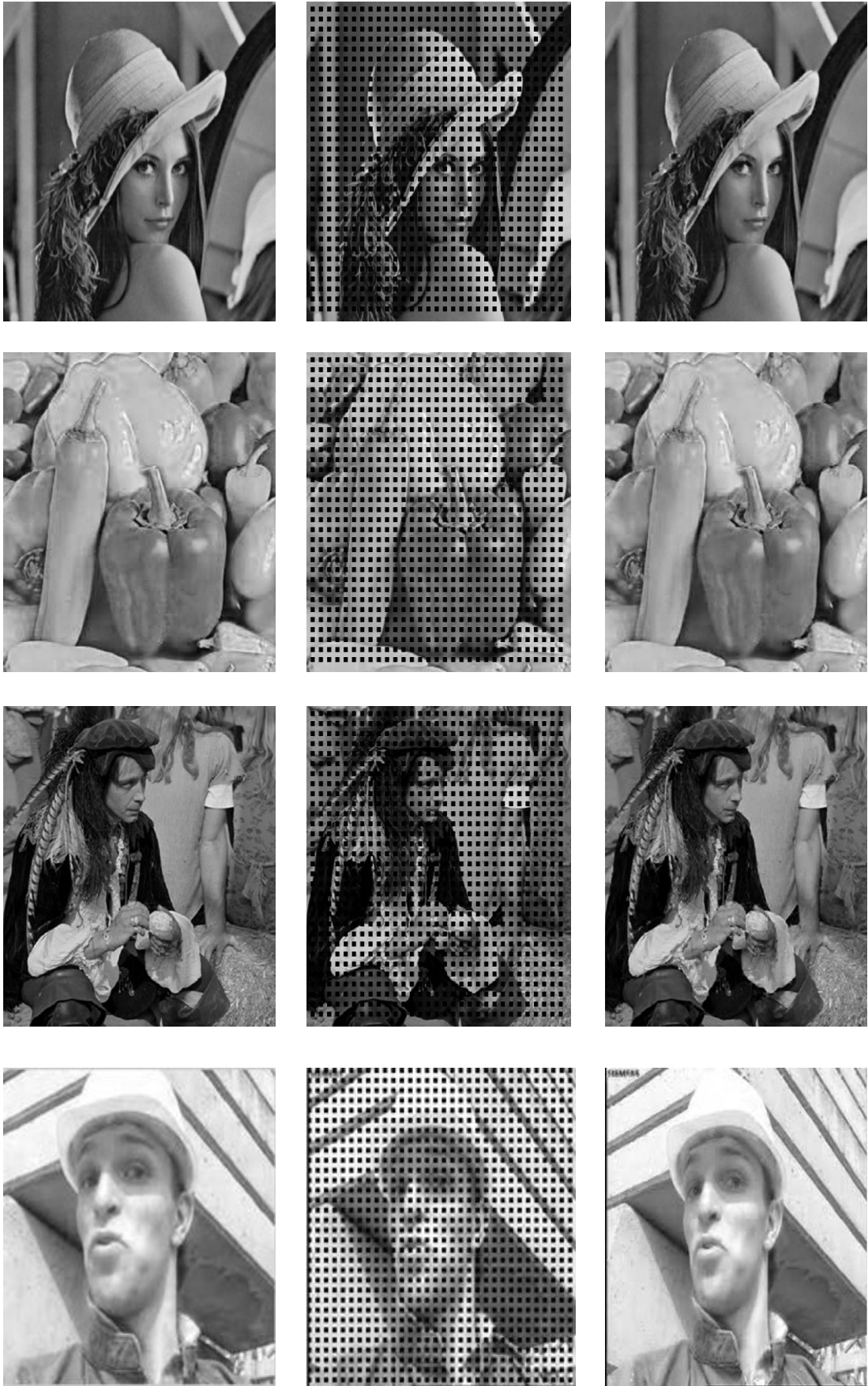


Figure 6.17: From left to right, the original images, the image with 25% missing MBs and the restored images for Lena, Peppers, Man and Foreman.

6.4.2 Regular Loss Pattern 16×16 Missing Block

A block loss size of 16×16 pixels is also applied to the proposed algorithm for further evaluation, Figure 6.18 demonstrates the subjective performance on Lena, Peppers, Man and Foreman images. In addition, Table 6.VIII shows the PSNR comparison between the proposed algorithm and previous works.

Table 6.VIII: Performance comparisons for MB loss size = 16×16 , PSNR calculated over the whole image for Lena and Peppers, with the Canny filter.

Methods	PSNR (dB)		
	<i>Lena</i>	<i>Peppers</i>	<i>Average</i>
Salama (Salama, Shroff & Delp, 1998)	35.01	34.71	34.86
Wang (Wang, Zhu & Shaw, 1993)	35.43	35.07	35.25
Sun (Sun & Kwok, 1995)	34.95	33.21	34.08
Park (Park <i>et al.</i> 2005)	35.98	35.50	35.74
Li (Li & Orchard, 2002)	37.48	38.27	37.87
Kim (Kim, Koo & Jeong, 2006)	37.37	38.95	38.16
Wavelet Edge-guided	37.65	38.97	38.31

Table 6.VIII shows objective comparison between the wavelet edge-guided interpolation method and six published works on the Lena and Peppers images. The outcomes show an improvement of the wavelet edge-guided method over the DCT edge-guided interpolation by 0.08 dB. In addition, it can be observed from the simulation results, that the average for the proposed wavelet edge-guided scheme has superior PSNR over the published works of Salama, Shroff & Delp (1998) by 3.45 dB, Wang, Zhu & Shaw (1993) by 3.06 dB, Sun & Kwok (1995) by 4.23 dB, Park *et al.* (2005) by 2.57 dB, Li & Orchard (2002) by 0.44 dB and Kim, Koo & Jeong (2006) by 0.15 dB.

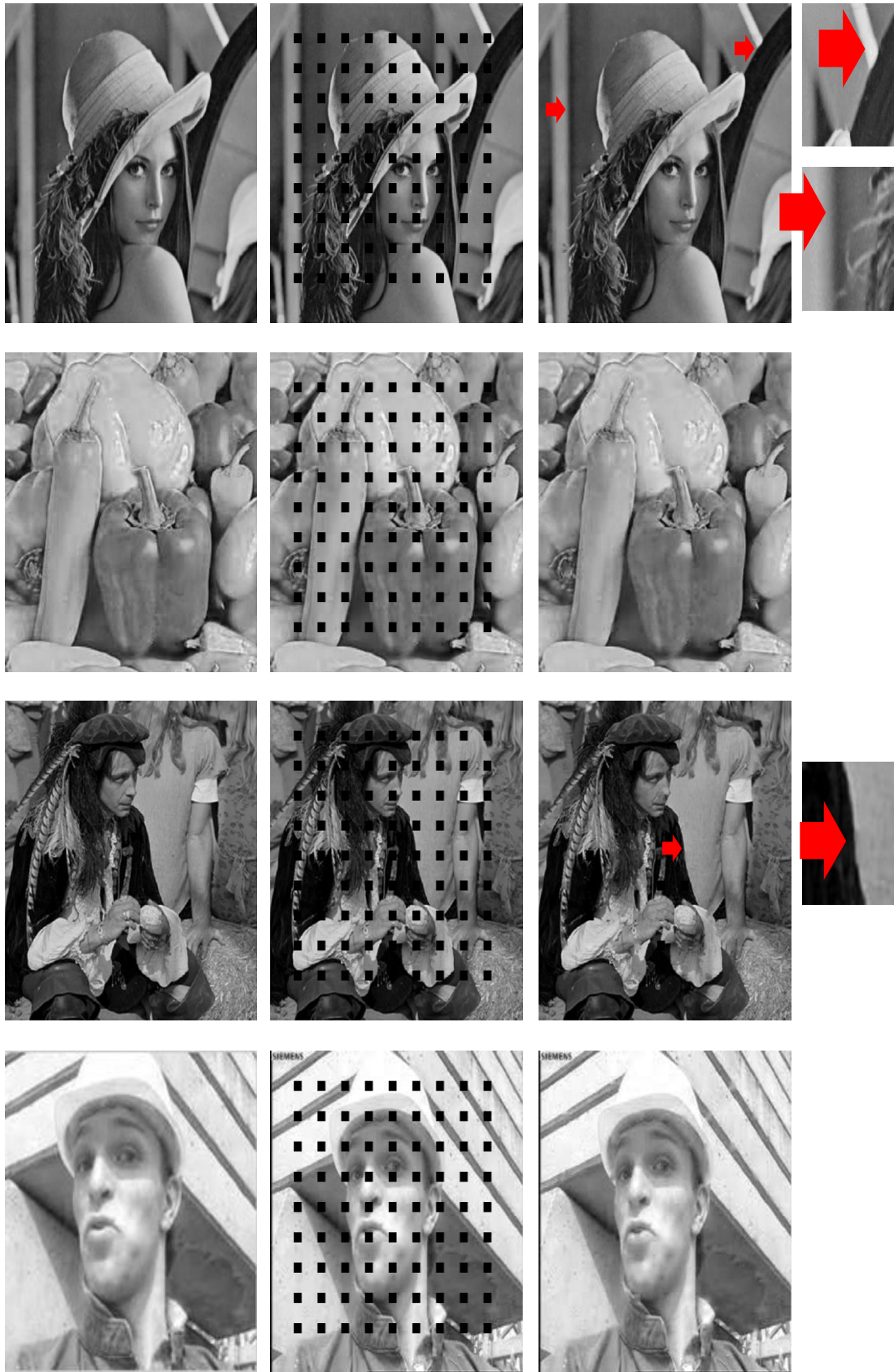


Figure 6.18: From left to right, the original images, corrupted images (16×16 isolated block loss), the restored images and zoomed in images for Lena, Peppers, Man and Foreman.

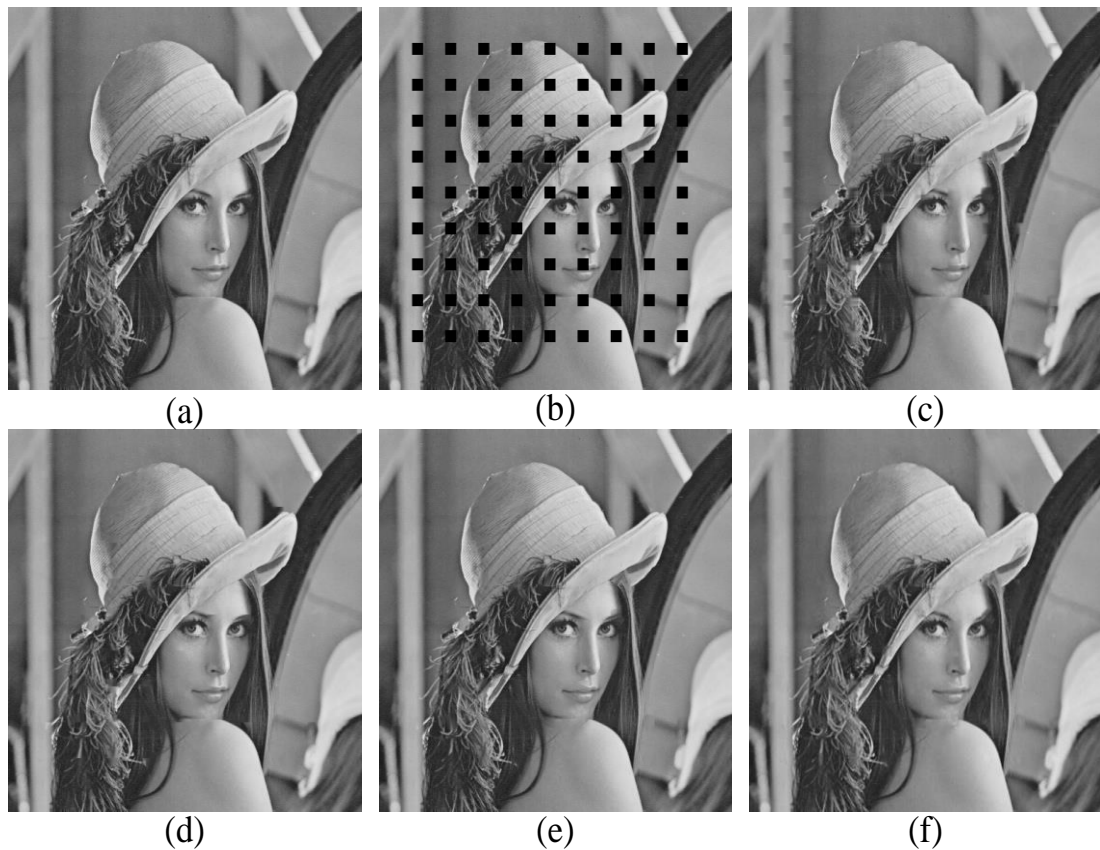


Figure 6.19: a) Original Lena image (512×512), b) Corrupted image (16 x 16 block loss). Images restored using the methods of (c) Sun & Kwok (PSNR = 34.95 dB), (d) Li & Orchard (PSNR = 37.41), (e) Kim, Koo & Jeong (PSNR = 37.37 dB) and (f) the proposed method (PSNR = 37.65 dB).

Figure 6.19 shows the performance comparison between the proposed method and three published works on the Lena image. The outcomes demonstrate that the proposed method improves the result by 0.28 dB compared with the best performance among all the other methods.

6.5 Evaluation for Random Loss Pattern

The random loss pattern is the same as in Chapter 5 and includes missing macro blocks that could be presented at any random position involving a random sequence of adjacent horizontal and/or vertical losses. In order to evaluate the performance result for the random block loss, the proposed method is applied to the Lena, Peppers, Man and Foreman images, degraded with random block loss, as shown in Figure 6.21.

The positions of the missing macro blocks are random and distinct in each evaluation test and therefore, the program is applied for a number of iterations to find the mean PSNR distortion, with this number being set to a value of ten for this part, as explained in subsection 5.2.3.2. Table 6.IX shows the number of iterations and the average for each of the ten values. The results are then compared with several published methods, including those for four published techniques that are taken from (Li & Orchard, 2002; Zhai *et al.*, 2010) directly along with that for the proposed method, being illustrated in Table 6.X. The PSNRs are averaged only for the missing pixels and hence, do not include those that are available inside the image. The results, after reconstruction, show that proposed method for the current research provides a better continuation of the structure across the missing areas.

Table 6.IX: Performance comparisons for a random MB loss rate of 10% (MB size = 8×8) on Lena with ten steps of iteration.

Image Result	Lena									
	1	2	3	4	5	6	7	8	9	10
PSNR (dB)	33.34	32.37	32.75	33.08	33.29	33.44	32.85	33.29	32.98	32.27
AVG.										
32.96										

As displayed in Table 6.X, the designed method outcome is better than the rest, and there is an improvement by 4.83 dB when compared with the best performance among all the prior results. The lowest outcome is that of Zeng & Liu (1995) which is 6.36 dB under what has been achieved here. Zhai *et al.* proposed two methods, the first being image error concealment via block-based bilateral filtering (Zhai *et al.*, 2008), and the second, Bayesian error concealment with a DCT Pyramid (Zhai *et al.*, 2010), which does achieve reasonably higher performance at 28.13 dB, but wavelet edge-guided surpasses both of these by 5.31 dB and 4.83 dB, respectively. Finally, Li & Orchard (2002) presented sequential error-concealment, whereby previously recovered pixels are used as input for the next stage of the recovery process and compared to the currently proposed method it has a lower PSNR of 5.58 dB.

Table 6.X: Performance comparisons for a random MB loss rate of 10%, MB size = 8×8, PSNR calculated just for the region of missing MBs for Lena, with the Canny filter.

Methods	Image Lena
	PSNR (dB)
Zhai <i>et al.</i> (2010)	28.13
Zhai <i>et al.</i> (2008)	27.65
Zeng & Liu (1995)	26.60
Li & Orchard (2002)	27.38
DWT edge-guided	32.96

The proposed method has been tested with a high level of random missing block, and Figure 6.20 shows the results for 25% and 40% random missing blocks on the Peppers image. Even though many missing blocks are included in each process, the proposed method is able to reconstruct the edges and texture within the image.

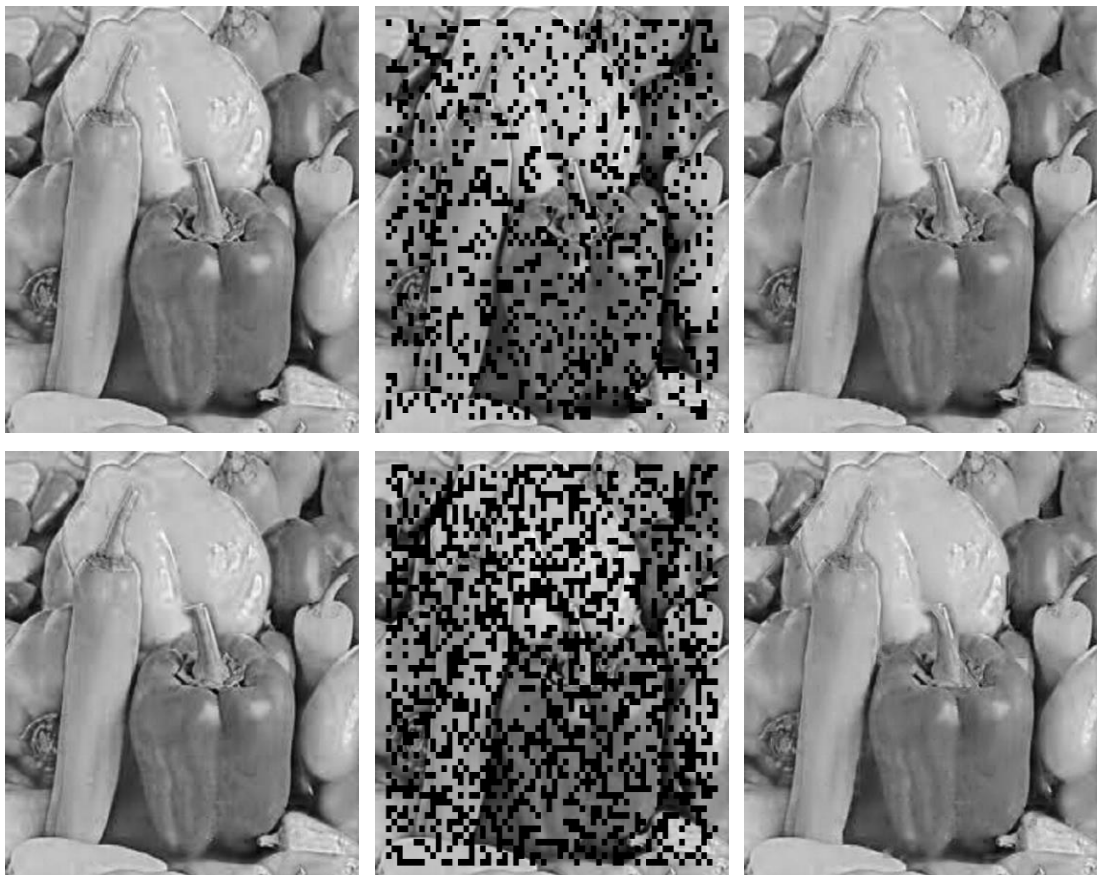


Figure 6.20: Top from left to right; the original Peppers, the image with 25% random missing MBs and the restored image and bottom the original Peppers, the image with 40% random missing MBs and the restored image.

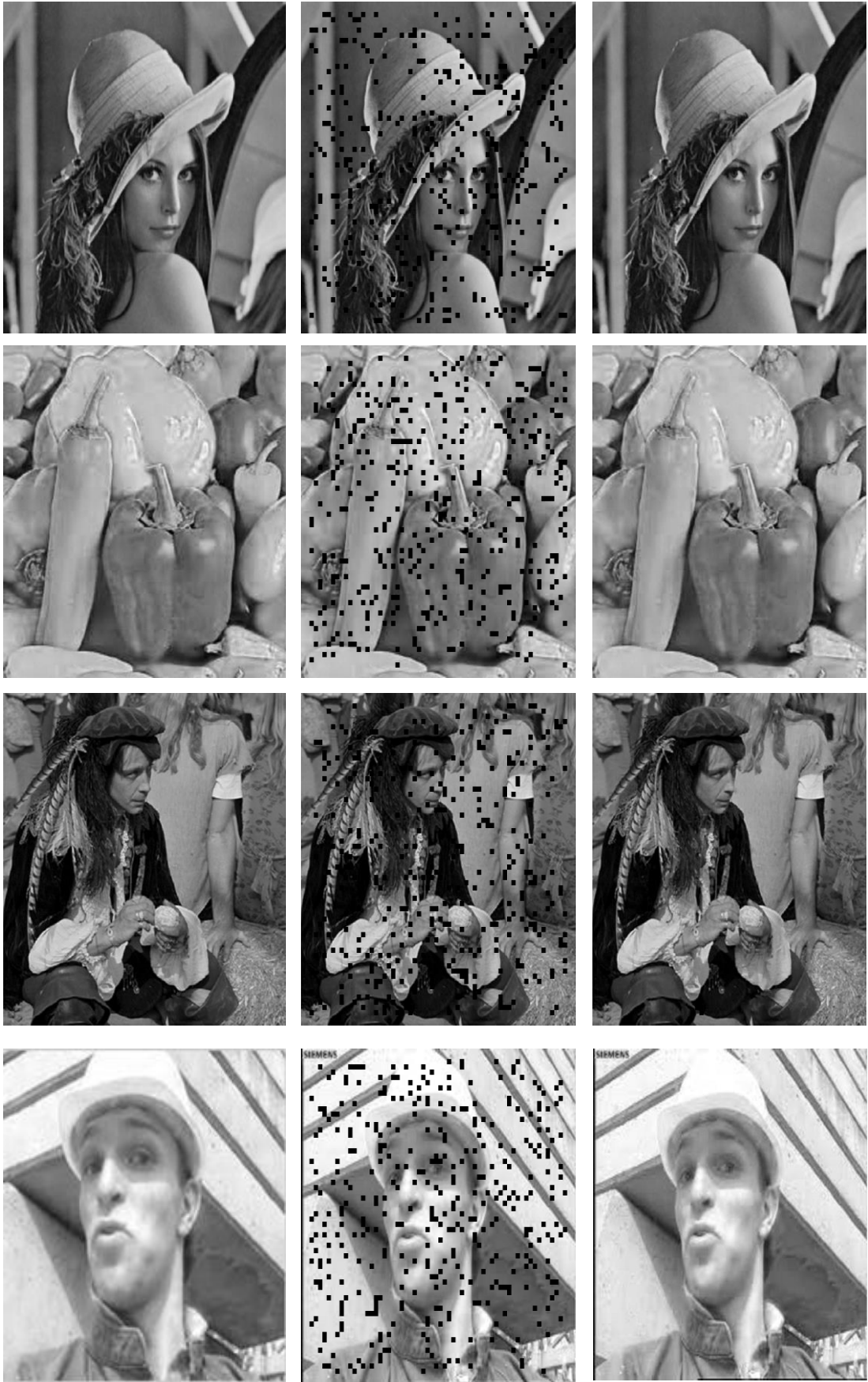


Figure 6.21: From left to right; the original Lena, Peppers, Man and Foreman images, with 10% random missing MBs and the restored images.

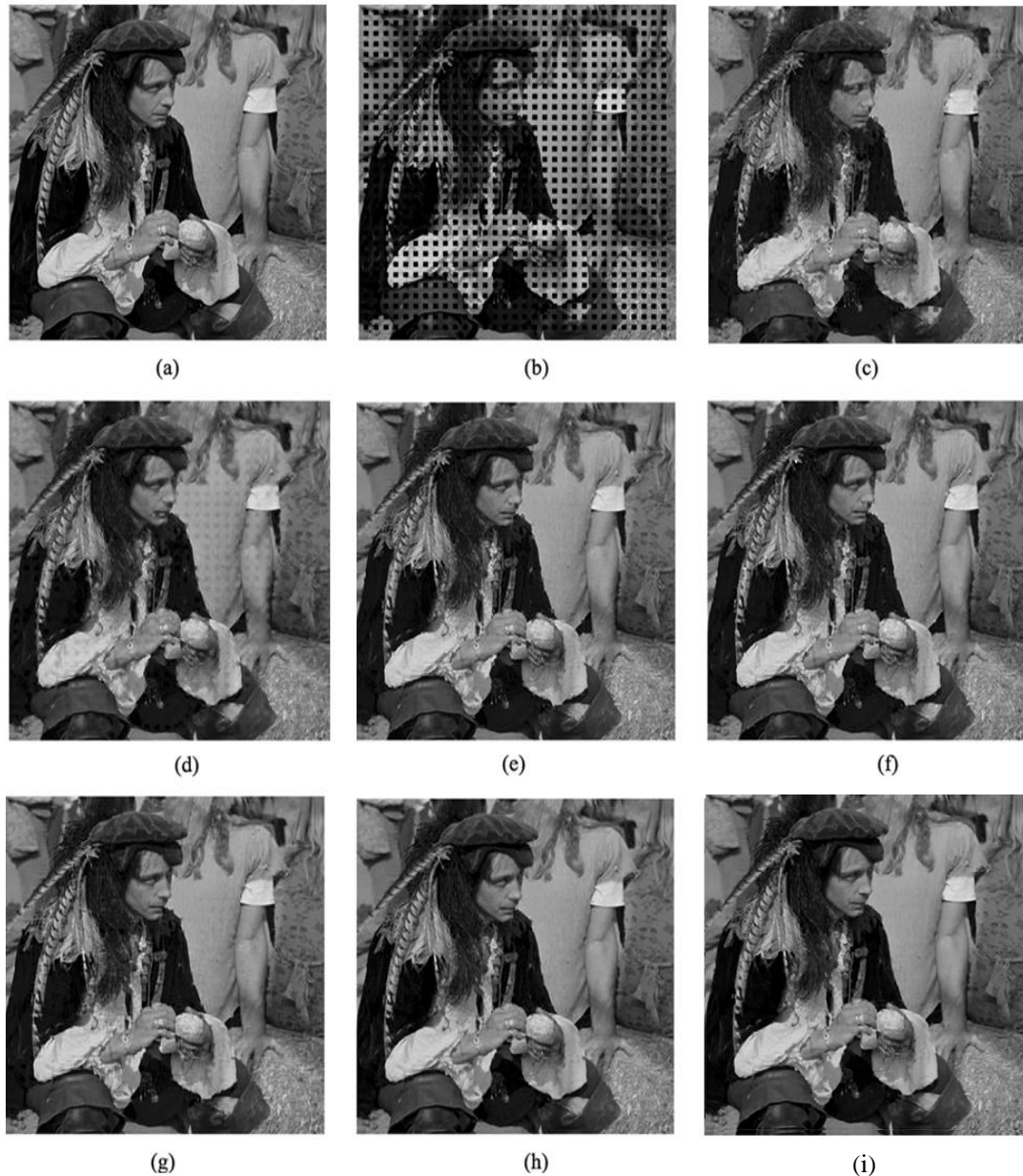


Figure 6.22: Experiment on a lost block size of 8×8 pixels of the “Man” image. (a) Original 512×512 and (b) damaged image of one missing block out of every four. Restoration using the methods of (c) Ancis and Giusto (PSNR = 25:47 dB), (d) Sun and Kwok (PSNR = 27:25 dB), (e) Hemami and Meng (PSNR = 27:65 dB), (f) Shirani *et al.* (PSNR = 27:44 dB), (g) Alkachouh and Bellanger (PSNR = 27:94 dB), (h) Park *et al.* (PSNR = 29:87 dB) and (i) the proposed method (31.61 dB).

Figure 6.22 illustrates the performance comparison between the proposed method and six published works on the Man image. The outcomes show that the proposed method improved the result by 1.74 dB when compared with the best performance among all the other methods. In addition, the suggested method is tested on a miscellaneous image, added specifically for this research (Figure 6.23).

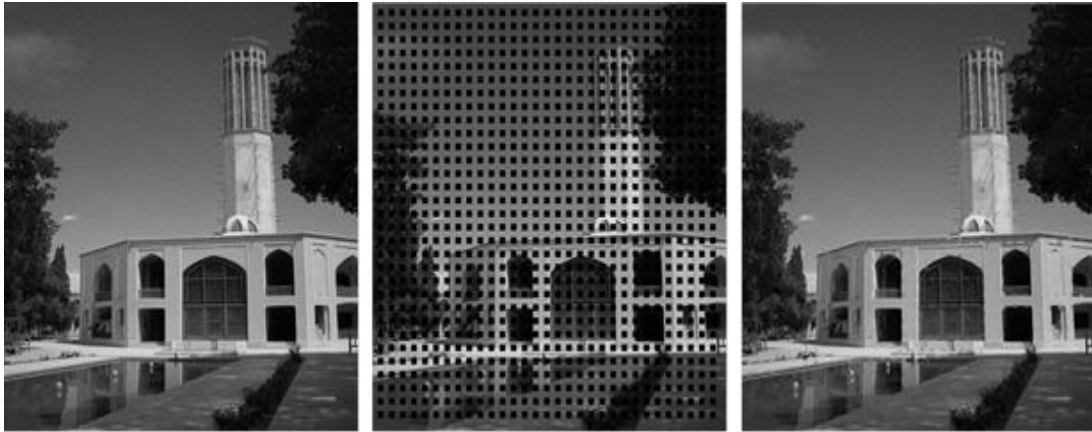


Figure 6.23: From left to right, original image, corrupted image (8×8 isolated block loss) and the restored image for the Dolat-Abad Garden, Yazd, Iran.

6.6 Run-Time Comparison

To compare the run time of different EC algorithms, test 512×512 images (*Lena*, *Baboon* and *Elaine*) are tested. The averaged run time for different loss pattern is presented in Table 6.X. The computation time reported in the table is obtained with non-optimized MATLAB implementations with Intel CORE i5, 2.3 GHz CPU and 4 GB memory.² We can see that the proposed algorithm is much faster (especially in the proposed DCT method) than the recently proposed [Liu *et al.* (2015), Koloda *et al.*, 2013, Li & Orchard, 2002, Koloda, Sanchez & Peinado, 2013] algorithms.

Table 6.X: Average run time in second comparison for 512×512 images, different MB loss rate for Lena, Baboon and Elain.

Methods	Run Time (second)		
	<i>Regular</i> 8×8	<i>Random</i> 8×8	<i>Random</i> 16×16
Sun & Kwok (1995)	4.82	1.91	2.59
Varsa, Hannuksela & Wang (2001)	0.10	0.08	0.07
Zhai <i>et al.</i> (2008)	1.28	0.57	0.59
Rongfu, Yuanhua & Xiaodong (2004)	4.59	2.81	9.09
Koloda <i>et al.</i> (2013)	79.75	29.84	124.68
Koloda, Sánchez & Peinado (2013)	426.59	170.88	89.16
Shirani, Kossentini & Ward (2000)	9.05	3.56	3.24
Li & Orchard (2002)	90.28	33.66	29.44
Liu <i>et al.</i> (2015)	155.30	63.22	53.30
Proposed DCT	53.62	43.32	45.71
Proposed DWT	65.19	56.98	58.13

Although the proposed algorithm requires longer time than some methods, its advantages over other methods are obvious in terms of objective and subjective evaluations, as shown in previous Sections. Experiments show there are slight improvements of details in DWT, compared with DCT, but the run time is higher for DWT (Table 6.X) as it was expected from previous works.

6.7 Conclusion

In this chapter, a novel multi-scale pyramid method using wavelet transform has been proposed as an alternative to DCT-pyramid image gap reconstruction. The proposed algorithm includes a combination of multi-resolution transforms, directional interpolation and edge-guided enhancement capable of restoring missing MBs including the edges. The experimental results demonstrate that improvement in the quality and PSNR of the restored images are obtained compared with the rest of proposed methods and in addition, these results are an improvement on the DCT edge-guided interpolation outcomes reported in Chapter 5.

This chapter has presented improvements in image gap restoration through incorporation of edge-based directional interpolation within a multi-scale pyramid transform. Which utilise three types of edges (a) the local edges or textures inferred from the gradients of the neighbouring pixels, (b) the global edges, or boundaries between image objects or segments, inferred using Canny edge detector application and (c) restored details information in cV, cH and cD.

The same methods of interpolation as the DCT based edge guided form are applied to the approximation section of the damaged image for wavelet based edge-guided interpolation. In addition, using the wavelet transform the values of three details parts (vertical, horizontal and diagonal) are represented and restored in the missing parts by using the available uncorrupted information. Finally, the two results are combined in order to achieve the final result, and it is found that the outcome of the process is improved and more efficient. Two possible alternative approaches to wavelet pyramid are used:

- (1) Wavelet decomposition and re-composition of the whole distorted images,

(2) Wavelet decomposition and re-composition of macro-block segmented images.

In the first approach, longer length wavelets tend to smear (mix) the boundaries of the distorted regions into those of the undistorted ones at successive levels of decomposition and hence cause degradation in the overall results. In order to overcome this when using higher length wavelets, macro block segmentation is applied to the image. In this method, the wavelet is applied to each macro block separately and consequently, distorted regions are not combined with undistorted ones at successive levels of decomposition.

In addition, to improving the result some unnecessary information in all the details coefficients parts that might cause false recovery in the restored image have been removed by hard-thresholding. Also, the iterative method in the last stage of using of the global edge interpolation produces a gain in PSNR. It starts the process by using a high level of Canny edge details at the first iteration and then continues the process by reducing the amount of these detail as a guide for the interpolation.

Chapter 7

Conclusion: Discussions and Future Works

Chapter 7

7. Conclusion: Discussion and Future Works

This chapter concludes the work by first discussing the appropriateness and the relative success of the selected pyramid frameworks and edge-guided interpolation methodologies presented in the different chapters of this thesis. The overall framework is based on the multi-scale pyramid transformation, and the methodologies that have been explored are the DCT-Pyramid, wavelets, texture interpolation and edge-guided interpolations. The discussion below covers the overall results obtained by the various methods and highlights the main contribution of the thesis. Finally, new directions for continuing the research work are proposed.

7.1 Conclusion

As explained, image loss in broadcast networks is due to a variety of different causes, such as congestion, network loss and signal fading. The impact of loss of data is a degradation of the image quality. Many interpolation methods have been proposed for estimating the missing image segments, with the aim of recreating an acceptable image quality, but problems with the output quality of restoration still persist and hence, there is scope for research leading to further improvements.

The main issues associated with existing gap interpolation methods relate to their deficiencies and inability to recover missing edges and texture details in gaps correctly. In particular, a reliable and effective interpolation method is required for ‘repairing’ the effect of missing blocks in still and moving images. In this thesis, a novel method for estimating unknown lost samples by interpolating from known

samples data has been derived. The concept presented is referred to as edge-guided pyramid image gap restoration.

The starting point of this research work was the selection, implementation and evaluation of an appropriate framework and methods for an image gap restoration technique. It was observed that using the EC interpolation method is the most common approach to gap restoration. Hence, the EC interpolation methods, which are based on the assumption that there is a high correlation between neighbouring pixels or macro blocks, were chosen as the focus of the current research.

The principal issues of the research were identified as follows:

- (1) *The choice of transformation domain*: pyramid transforms were selected for transformation as they have two desirable properties, (i) transformation to various sets of frequencies and scales and (ii) progressive reduction of the gap size to one sample wide at the pyramid apex. Furthermore, two types of pyramid transforms were explored; namely DCT-pyramid and wavelets.
- (2) *The choice of interpolations*: a family of interpolation methods including; weighted linear, nearest neighbour, cubic and spline were explored for texture and edge-guided interpolations.
- (3) *The choice of edge detection methods*: different edge detection methods, Sobel and Canny, were comparatively explored.
- (4) *Gap blending and post-processing methods*: developed as an integral part of each stage processes, such as DCT based blending, to improve the results further.
- (5) *Evaluations*: the methods were comparatively evaluated with a range of widely used degradations, performance measures and test images.

In order to address the image enhancement problems successfully the novel incorporation of ‘local’ and ‘global’ edge enhancement strategies within a pyramid DCT/DWT image processing framework has been proposed. The algorithm includes a combination of multi-resolution DCT/DWT transforms, directional interpolation and edge-guided enhancement capable of restoring missing MBs including the edges.

The proposed method utilises two types of edges in the DCT-based techniques:

- a) The local edges or textures inferred from the gradients of the neighbouring pixels;
- b) The global edges, or boundaries between image objects or segments, inferred using Canny edge detectors.

For the DWT-based method, the propose method, in addition to utilising the local and global edges (as in *a* and *b* above) also involves using:

- c) Restored details information in *cV*, *cH* and *cD*.

where, *cV*, *cH* and *cD* are the vertical, horizontal and diagonal details of the wavelets, respectively. The experimental results demonstrate that significant improvement in the quality and PSNR of the restored images are obtained by combining local and global edge-guided image restoration.

The main contributions of this research work are as follows:

1. Incorporation of edge-guided interpolation within multi-scale pyramid transformation for image gap restoration. A baseline DCT-based pyramid transformation incorporates ‘local’ and ‘global’ edges within images.
2. The use of initial texture interpolation for subsequent local edge-guided pyramid interpolation for image gap restoration. The proposed interpolation algorithms are aimed at providing a gap estimate consistent with the local edges derived from the gradient of the known neighbouring pixels surrounding the gap. However, the results obtained were not satisfactory.
3. Further improvement investigated via the inclusion of global edge estimates and interpolators within the pyramid transforms image restoration. This involves the use of edge-detection and segmentation of the image objects. Subsequently, the interpolation is confined within homogenous regions to avoid blurring across edges and loss of edge information.

4. Comparative evaluation of the incorporation of Sobel and Canny detectors within the pyramid interpolation. For the Canny detector, its parameters, namely, the variance of the Gaussian filter and the threshold of the significant edges, were experimentally optimised. Moreover, an iterative restoration methodology was applied to the last stage of interpolation.
5. Exploring a family of wavelets as an alternative to DCT pyramid transforms for edge-guided image gap restoration. Further, three corrupted details parts of wavelets (cV, cH and cD) have been restored, as these include the edge information, the restored details combine with the restored approximation coefficients.

In addition to the main contributions above, the following processing algorithms have been investigated.

Pyramid Transforms: DCT versus Wavelets

Research into a pyramid structure, that lends itself to the use of the DCT or various families of DWT as the kernel function, has been investigated. The main justification for the choice of a pyramid as the framework for interpolation is pragmatic; a relatively large gap at the base of the pyramid is reduced to a single sample at the apex, which can be conveniently interpolated and propagated to the next level of interpolation. The process is then reversed and at each stage, the missing samples are estimated, up-sampled and combined with the available samples. Multi-scale transformation into pyramid layers facilitates interpolation and estimation of missing gaps by creating an information pyramid.

Subsequent work on gap interpolation introduced in this thesis has included the use of complex wavelets instead of DCT for pyramid decomposition. A novel multi-scale pyramid method using wavelet transform has been proposed as an alternative to DCT-pyramid image gap reconstruction. The proposed algorithm includes a combination of multi-resolution transforms, local directional interpolation and edge-guided enhancement capable of restoring missing MBs including the edges. The

experimental results demonstrate an improvement in the quality and PSNR of the restored images that are obtained compared with published methods. Furthermore, the wavelets based results are better than those for DCT edge guided interpolation.

The same methods of interpolation as the DCT based edge-guided interpolation have been applied to the approximation section of the damaged image for wavelet based edge-guided interpolation. By using the wavelet transform the values of three details parts (vertical, horizontal and diagonal) are represented and are restored by using the available uncorrupted information in the vicinity of the missing parts. The result is then enhanced by applying the edge guided interpolation. Finally, the approximate and details results are combined to achieve the outcome of the process which proves to be better in quality and efficiency.

Two alternative approaches to wavelet pyramid are used:

- (1) Wavelet decomposition and re-composition of the whole distorted images;
- (2) Wavelet decomposition and re-composition of macro-block segmented images.

By applying the first approach, longer length wavelets tend to smear (mix) the boundaries of the distorted regions into those of the undistorted ones at successive levels of decomposition and hence, cause degradation in the overall results. In order to overcome this when using longer length wavelets, macro block segmentation is applied to the image, then wavelet being applied to each macro block separately. Consequently, distorted regions are not combined with undistorted ones at successive levels of decomposition. The investigation on various types of wavelets transforms shows that the best wavelet type for the proposed method is Daubechies version 1, abbreviated db1.

Moreover, to improve the result some unnecessary information in all three details coefficients parts, which might cause false recovery in the restored image, have been removed by hard-thresholding.

Alternative Methods of Interpolation and Blending

Another important factor is choosing the best interpolation method at each level of the pyramid. Hence, the impact of using four different interpolation methods for gap estimation was assessed, these being the: nearest neighbour, linear, spline and cubic techniques. The interpolators were first compared based on their comparative ability to retrieve a down-sampled image, without any gap, for down-sampling rates 2, 4, 8. The results in section 4.4, show that spline interpolation performs best followed by the cubic, linear and nearest neighbour interpolators. However, as the down-sampling rate and hence, information loss rate increases, the difference in the performance of the nearest neighbour interpolator (the worst) and the spline interpolator (the best) decreases. Similarly, for the case where there is loss of image macro blocks, then the differences in the performance of the various interpolators decrease.

In order to improve the results further, the combination of interpolation and post processing, blending, functions have been performed with two different estimators (mean and median) at the apex of the pyramid. As expected, the median approach achieves better results as it is an edge-preserving statistic. The first observation is that all interpolation methods employed within pyramid estimation, result in reasonably high and very similar values of PSNR of around 31 dB and an SSIM of 0.92. It is proposed that the reasons for the similar results are:

- 1) The relatively high correlation that exists among neighbouring image pixels;
- 2) The efficiency of the pyramid DCT/DWT structure in capturing the correlations of the image pixels;
- 3) The gap loss, since the experimental result indicates that as the information loss increases the interpolators' performances converge to similar values.

As a further processing step, to blend the estimate of a block within its surrounding texture, a DCT based mixing akin to a form of data-dependent low-pass filtering has been used.

Methods for Texture Interpolation

As an improvement to the up-sampling and blending method of gap restoration, the use of local edges is explored. The local edges have been inferred from the available neighbouring pixels of the missing areas without the benefit of global edge detectors, which has allowed for capture of the texture information. At the apex of the pyramid the local directional interpolation preserves three types of local edges (horizontal (H), vertical (V), cross (C)), which cover all eight surrounding directions of the missing pixels.

Then in next stages of the pyramid approach, the missing pixels are progressively replaced from the outer boundaries towards the block centre by the local edge-guided interpolation by utilising edges in the horizontal, vertical and cross directions. In general, the interpolation inferred from four possible edge directions is combined with the pixel estimate obtained from the previous stage so as to compute a new estimate of the missing block.

The performance of the local edge-guided interpolation is increased compared with the simple up-sampling method.

Alternative Methods for Edge Detection

After using the local edge-guided interpolation, the next process is to apply the global edge-guided interpolation. The global edges, or boundaries between image objects or segments, are inferred using different types of edge detector applications.

Edge detection methods are explored in this research to improve the image quality further. It has been concluded that in the specified applications the edge detection methods are structurally a combination of image smoothing and image differentiation followed by post-processing for edge labelling. The smoothing reduces the influence of noise and random fluctuations. Image differentiation provides information on intensity transition in the image that is necessary to represent the position and strength of the edges and their orientation. The edge labelling requires post-processing to suppress the false edges, linking the dispersed ones, and producing a uniform contour of objects.

As the gap restoration performance is hampered by edge detector errors, finding the best operator is vital in order to achieve the best performance. The outcomes of various edge detection methods are investigated and show that the best result belongs to the Canny edge detector, whilst Sobel and Prewitt are approximately the same and the worst result comes from the Roberts edge detection. Hence, the two most prominent edge detectors, Sobel and Canny, have been implemented in this research (Chapter 5) and the results demonstrate that the Canny edge detection achieves a higher outcomes as it was expected from the literature.

The parameters (variance and threshold) of the Canny edge detector need to be optimised in order to achieve the best result. In addition, there is some unnecessary information in edge-based segmentation of images that do not represent significant edges. The insignificant edge data might cause false recovery in the restored image and therefore, need to be removed. The method designed for this strives to retain significant image edge information and eliminate the remainder. Hard-thresholding is used in the proposed algorithm that involves checking each edge intensity and setting values less than or equal to the threshold of zero.

Interpolation in an Iterative Loop

The experiments concluded in this research show that using an iterative method in the last stage of pyramid estimation produces a gain in PSNR. At the base level of the process, an iterative edge pruning strategy is applied for edge detection. This relies on varying the two parameters of a Canny detector, the variance of the Gaussian filter and the threshold of the significant edges, at each iteration in order to achieve improved results.

The experiment starts the process of edge detection and interpolation, by using a higher level of Canny edge details at the first iteration and then reduces the amount of the details. The simulation results, show the overall PSNR obtained by fixing the Gaussian filter variance at an empirically obtained optimal value and then varying the threshold in the range 0.01-0.05. Note that starting from a threshold value of 0.01 the best PSNR is obtained at the 4th iteration after three discrete -step increases in the

threshold value. Hence, the optimal number of iteration for this research is four as before this the result has an upward trend and subsequent to it, levels off.

Discussions of Overall Results

The methods proposed in this thesis were evaluated over a range of test images with a range of different loss rates.

This involved evaluating the performances of various different proposed methods of interpolation for an MB loss rate of 25% and MB size of (8×8) on five well-known test images, where the PSNR was calculated for the whole image. The pyramid wavelet using Canny edge detector offered the best average result (at 34.40 dB), followed by pyramid DCT Canny (at 34.34 dB). The worst result is from simple up-sampling (at 28.01 dB), which only utilises an interpolation method to up-sample from one layer to the next. The outcomes of the proposed method show that applying a combination of local and global edge detection improves the interpolation along with use of pyramid wavelet transformation.

The same results were obtained for an MB loss rate of 25% and MB size of (8×8) on Lena, where the PSNRs were calculated for just the missing parts. The best outcome among thirteen published works belongs to the pyramid wavelet using a Canny edge detector (at 29.83 dB). In addition, in case of calculating the PSNRs over whole image, another thirteen techniques are compared and the best outcome belongs to the proposed method at 36.08 dB.

Moreover, the performance of the proposed methods was assessed on a random MB loss rate of 10% (MB size= 8×8), where the PSNRs was calculated just for the region of the missing block on Lena. The results show that by adding each level of processing, the PSNR is improved and it is concluded that the proposed pyramid wavelet Canny method (at 32.96 dB) provides the best PSNR and visual image quality, followed by pyramid DCT Canny and pyramid DCT Sobel at 32.80 dB and 30.78 dB respectively. Finally, the performance of pyramid DCT Local edge is (at 30.05 dB).

It is concluded that by applying the proposed interpolation the outcome improves and the best restored image quality is achieved.

Achievements and Limitation:

In this research a method is proposed for restoration of lost macro-blocks in digital images. The proposed algorithm includes combination of multi-resolution transforms, directional interpolation and edge-guided enhancement capable of restoring missing blocks including the edges. The main contribution of this work is the incorporation of local and global edge-guided interpolators within a pyramid structure in an iterative loop at the last stage.

Two types of pyramid transformation were evaluated namely DCT and wavelets. The methods were evaluated on a number of different test images in a range of loss rates for regular and random pattern of losses. The results for DCT and wavelets are similar (with a slight improvement of details in DWT) and achieve better performance than other state-of-the-art methods in terms of objective and subjective evaluation. The incorporation of local, global edges and iterative process improves interpolation. The results obtained from DCT pyramid are comparable with those obtained from wavelets with the DWT offering a slight advantage in computation time. The experimental results demonstrate that significant improvement in the quality and PSNR of the restored images are obtained by the proposed edge guided image restoration method. An interesting aspect of this work is the use of iterative methods for improving various layers of pyramid reconstruction including the image and the edge, or skeleton, layers.

Although the proposed algorithm requires longer time than some methods, its advantages over other methods are obvious in terms of objective and subjective evaluations, as shown in previous Sections.

There is a limitation in the proposed methodology, when the macro block loss exceeds the range of 60% in random loss pattern, the method is not able to recover the image.

Future Works

Proposed future works include alternative families of pyramid transforms, different edge detection methodologies and other interpolation approaches.

An alternative pyramid that could be explored is complex Gabor. It has been used in many image analysis applications thus, makes it a popular method for feature extraction.

For further work on edge detection, within the pyramid framework, the details components of wavelets, (namely cH, cA, cD) might provide more useful basis for the integration of wavelet analysis and edge detection, such that edges or image skeletons are a by-product of the wavelet analysis. This could involve devising strategies regarding the fusion of information provided by the wavelets details at various layers of the pyramid.

Implementation of the proposed methods for real-time embedded applications is another potential avenue for research. This will involve tailoring the different methodologies for various platforms and, depending on the required constraints, fast and power efficient execution.

An interesting aspect of this work is the use of iterative methods for improving the quality of various layers of pyramid reconstruction including the image, the edge or the skeleton, layers. This is a further area of research where work may be fruitful.

The choice of interpolation methods is a further area that could be the subject of future research. The structure in a pyramid can be used such that the interpolation at each layer is informed (e.g. via a Bayesian formulation) from the results obtained in the preceding layers.

A further area of for investigation is the incorporation of the pyramid methodology for video applications. This might involve the use of motion detection and the progressive efficient updating of the pyramid information as the scenes and contents in successive image frames evolve.

In addition to image restoration, the incorporation of a pyramid structure, with edge detection and edge-guided interpolation, might lend itself to other areas of image processing, such as efficient multi-layered image coding that can provide different levels of service depending on the bandwidth available.

Last but not least more work can be done for investigating on the computational complexity to make the proposed method more efficient.

References

- Machu Picchu: the highest resolution image ever taken* 2016, [Homepage of telegraph.co.uk], [Online]. Available: <http://www.telegraph.co.uk/travel/activityandadventure/9681432/Machu-Picchu-the-highest-resolution-image-ever-taken.html>.
- Mobile Devices: Nokia Lumia 1020* 2016, [Homepage of Microsoft], [Online]. Available: <https://www.microsoft.com/en-gb/mobile/phone/lumia1020/>.
- Adelson, E.H., Anderson, C.H., Bergen, J.R., Burt, P.J. & Ogden, J.M. 1984, "Pyramid methods in image processing", *RCA Engineer*, vol. 29, no. 6.
- Agrafiotis, D., Bull, D.R. & Canagarajah, C.N. 2006, "Enhanced error concealment with mode selection", *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 16, no. 8, pp. 960-973.
- Ahmed, N., Natarajan, T. & Rao, K.R. 1974, "Discrete Cosine Transform", *Computers, IEEE Transactions on*, vol. C-23, no. 1, pp. 90-93.
- Aign, S. & Fazel, K. 1995, "Temporal and spatial error concealment techniques for hierarchical MPEG-2 video codec", *Communications, 1995. ICC '95 Seattle, 'Gateway to Globalization', 1995 IEEE International Conference on*, pp. 1778.
- Albanese, A., Blomer, J., Edmonds, J., Luby, M. & Sudan, M. 1996, "Priority encoding transmission", *Information Theory, IEEE Transactions on*, vol. 42, no. 6, pp. 1737-1744.
- Alejandro A. Ramírez-Acosta, A.A., Mireya S. García-Vázquez, S. Mireya & Nagano, M. 2012, "Adaptive Spatial Concealment of Damaged Coded Images", *Springer, MCP*, vol. 7329, pp. 94-106.
- Alkachouh, Z. & Bellanger, M.G. 2000, "Fast DCT-based spatial domain interpolation of blocks in images", *Image Processing, IEEE Transactions on*, vol. 9, no. 4, pp. 729-732.
- Ancis, M. & Giusto, D.D. 1999, "Reconstruction of missing blocks in JPEG picture transmission", *Communications, Computers and Signal Processing, 1999 IEEE Pacific Rim Conference on*, pp. 288.
- Asghari Oskoei, M. & Hu, H. 2010, "A Survey on Edge Detection Methods," *Technical methods*, School of Computer Science & Electronic Engineering University of Essex.

- Asheri, H., Rabiee, H.R., Pourdamghani, N. & Ghanbari, M. 2012, "Multi-directional spatial error concealment using adaptive edge thresholding", *Consumer Electronics, IEEE Transactions on*, vol. 58, no. 3, pp. 880-885.
- Bajic, I.V. & Woods, J.W. 2003, "Domain-based multiple description coding of images and video", *Image Processing, IEEE Transactions on*, vol. 12, no. 10, pp. 1211-1225.
- Banham, M.R. & Katsaggelos, A.K. 1997, "Digital image restoration", *Signal Processing Magazine, IEEE*, vol. 14, no. 2, pp. 24-41.
- Basu, M. 2002, "Gaussian-based edge-detection methods-a survey", *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 32, no. 3, pp. 252-260.
- Belfiore, S., Grungetto, M., Mugli, E. & Olmo, G. 2003, "An error concealment algorithm for streaming video", *IEEE, Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on (Volume:3)*, 14-17 Sept. 2003, pp. 649.
- Belfiore, S., Grangetto, M., Magli, E. & Olmo, G. 2005, "Concealment of whole-frame losses for wireless low bit-rate video based on multiframe optical flow estimation", *Multimedia, IEEE Transactions on*, vol. 7, no. 2, pp. 316-329.
- Boliek, M., Christopoulos, C. & Majani, E. 2000, "JPEG2000 Part I Final Draft International Standard", vol. (ISO/IEC FDIS15444-1), no. ISO/IEC JTC1/SC29/WG1 N1855.
- Bonatz, D. 2007, , *The Ancient Near East: A History of Images*. Available: http://www.geschkult.fu-berlin.de/en/e/vaa/projekte/History_of_Pictures.
- Burt, P.J. & Adelson, E.H. 1983, "The Laplacian Pyramid as a Compact Image Code", *Communications, IEEE Transactions on*, vol. 31, no. 4, pp. 532-540.
- Canny, J. 1986, "A Computational Approach to Edge Detection", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. PAMI-8, no. 6, pp. 679-698.
- Chang, E. 1998, "An Image Coding and Reconstruction Scheme for Mobile Computing", 5th IDMS, Oslo, Norway, September, pp. 137.
- Coifman, R.R. & Wickerhauser, M.V. 1992, "Entropy-based algorithms for best basis selection", *Information Theory, IEEE Transactions on*, vol. 38, no. 2, pp. 713-718.
- Curtin, D.P. , *A Short Course Book Curtin's Guide to Digital Cameras and Other Photographic Equipment*. Available: <http://www.shortcourses.com/guide/guide1-3.html>.

- Debrunner, V., Debrunner, L., Wang, L. & Radhakrishnan, S. 2000, "Error Control And Concealment For Image Transmission", *IEEE Communications Surveys & Tutorials*, [Online], vol. 3, no. 1. Available from: <http://www.comsoc.org/pubs/surveys>.
- Ding, J.J., Wei, W.Y. & Chen, H.H. 2011, "Context-based adaptive zigzag scanning for image coding", *Visual Communications and Image Processing (VCIP), 2011 IEEE*, pp. 1.
- Dony, R.D. 2001, "Chapter 1:Karhunen-Loève Transform" in *The Transform and Data Compression Handbook* Boca Raton, CRC Press LLC, .
- Dorini, L.B. & Leite, N.J. 2009, "Multiscale Methods for Image Processing: The Wavelet and the Scale-Space Approaches", *Computer Graphics and Image Processing (SIBGRAPI TUTORIALS), 2009 Tutorials of the XXII Brazilian Symposium on*, pp. 31.
- Feamster, n. & Balakrishnan, h. 2002, "Packet Loss Recovery for Streaming Video", *M.I.T. Laboratory for Computer Science, 12th International Packet Video Workshop*, .
- Floyd, J.S., Padhye, J. & Widmer, J. 2000, "Equation-Based Congestion Control for Unicast Applications", *SIGCOMM'00, Proceedings of the conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, August, pp. 43.
- Getreuer, P. "Linear Methods for Image Interpolation", *Image Processing On Line*, [Online], . Available from: http://dx.doi.org/10.5201/ipol.2011.g_lmii.
- Gharavi, H. & Gao, S. 2008, "Spatial interpolation algorithm for error concealment", *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, pp. 1153.
- Girotra, A. , *Motion Estimation an Overview* [Homepage of Trainee Design Engineer], [Online]. Available: <http://www.docfoc.com/motion-estimation-an-overview-by-abhishek-girotra-trainee-design-engineer>.
- Gonzales, R.C. & Woods, R.E. 2008, *Digital Image Processing* , 3rd edn, Pearson.
- Greenspun, P. , *History of Photography Timeline* [Homepage of June 1999], [Online]. Available: <http://photo.net/history/timelin>.
- Haberdar, H. 2012, , *Discrete Cosine Transform Tutorial*. Available: www.haberdar.org.

- Hayasaka, M., Gamage, M. & Miki, T. 2005, "An efficient loss recovery scheme for on-demand video streaming over the internet", *Advanced Communication Technology, 2005, ICACT 2005. The 7th International Conference on*, pp. 1301.
- He, L. & Zhang, Y. 2010, "A Rotate-based Best Neighborhood Matching Algorithm for High Definition Image Error Concealment", *Computer and Information Technology (CIT), 2010 IEEE 10th International Conference on*, pp. 1393.
- Hemami, S.S. 1995, "Digital Image Coding For Robust Multimedia Transmission", *Symp. Multimedia communication and video coding* New York.
- Hemami, S.S. & Gray, R.M. 1997, "Subband-coded image reconstruction for lossy packet networks", *Image Processing, IEEE Transactions on*, vol. 6, no. 4, pp. 523-539.
- Hemami, S.S. & Meng, T.H.-. 1995, "Transform coded image reconstruction exploiting interblock correlation", *Image Processing, IEEE Transactions on*, vol. 4, no. 7, pp. 1023-1027.
- Hsia, S.C. 2004, "An edge-oriented spatial interpolation for consecutive block error concealment", *Signal Processing Letters, IEEE*, vol. 11, no. 6, pp. 577-580.
- Ira, M. 2008, *Polycom's Lost Packet Recovery (LPR) Capability*, Wainhouse Research, polycom.
- Jain, J. & Jain, A. 1981, "Displacement Measurement and Its Application in Interframe Image Coding", *Communications, IEEE Transactions on*, vol. 29, no. 12, pp. 1799-1808.
- Jawerth, B. & Sweldens, W. 1993, "An Overview of Wavelet Based Multiresolution Analyses", vol. 36, pp. 377-412.
- Jonic, S. & Soriano, C.O.S. 2011, "Splines in biomedical image processing" in *Optical and Digital Image Processing: Fundamentals and Applications* Wiley, , pp. 119-134.
- JPEG , *Overview of JPEG 2000*. Available: <http://www.jpeg.org/>.
- Jung, K.J., Chang, J. & Lee, C. 1994, "Error concealment technique using projection data for block-based image coding", *SPIE 2308, Visual Communications and Image Processing '94, 1466*, pp. 1466.
- Katzmann, m. , *The history of the photogravure*. Available: http://www.photogravure.com/history/chapter_niepce.html.
- Kaup, A., Meisinger, K. & Aach, t. june 2005, "Frequency selective signal extrapolation with applications to error concealment in image communication",

AEU-International Journal of electronics and communication, vol. 59, no. 3, pp. 147-156.

- Khaziakhmetov, M. & Zakharova, T. 2012, "The using of wavelet analysis in climatic challenges", *Stochastic Models and their Applications Faculty of Informatics, Annales Mathematicae et Informaticae*, University of Debrecen, August 22–24, pp. 159.
- Khoshelham, K. & Elberink, S.O. 2012, "Accuracy and Resolution of Kinect Depth Data for Indoor Mapping Applications", vol. 12, pp. 1437.
- kim, B. 2002, *Numerical Optimization Methods for Image Restoration*, Stanford university.
- Kim, K.S., Lee, H.Y. & Lee, H.K. 2010, "Spatial error concealment technique for losslessly compressed images using data hiding in error-prone channels", *Communications and Networks, Journal of*, vol. 12, no. 2, pp. 168-173.
- Kim, W., Koo, J. & Jeong, J. 2006, "Fine directional interpolation for spatial error concealment", *Consumer Electronics, IEEE Transactions on*, vol. 52, no. 3, pp. 1050-1056.
- Klein, A.S. 1990, "High Resolution and Image Compression Using the Discrete Cosine Transform", *SPIE Vol. 1249 Human Vision and Electronic Imaging: Models, Methods, and Applications*, October.
- Kolås, Ø., *Image Processing with gluas*. Available: http://pippin.gimp.org/image_processing/index.html.
- Koloda, J., Ostergaard, J., Jensen, S.H., Sanchez, V. & Peinado, A.M. 2013, "Sequential Error Concealment for Video/Images by Sparse Linear Prediction", *Multimedia, IEEE Transactions on*, vol. 15, no. 4, pp. 957-969.
- Koloda, J., Peinado, A.M. & Sanchez, V. 2013, "On the application of multivariate kernel density estimation to image error concealment", *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pp. 1330.
- Koloda, J., Sánchez, V., & Peinado, A. M. 2013, "Spatial error concealment based on edge visual clearness for image/video communication," *Circuits, Syst., Signal Process.*, vol. 32, no. 2, pp. 815–824.
- Kumar, S. 2001, 22.10.2001-last update, *An Introduction to Image Compression*. Available: <http://www.debugmode.com/imagecmp/>.
- Kwok, W. & Sun, H. 1993, "Multi-directional interpolation for spatial error concealment", *Consumer Electronics, IEEE Transactions on*, vol. 39, no. 3, pp. 455-460.

- Lee, L.W., Wang, J.F., Lee, J.Y. & Shie, J.D. 1993, "Dynamic search-window adjustment and interlaced search for block-matching algorithm", *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 3, no. 1, pp. 85-87.
- Lee, P.J. & Chen, L.G. 2002, "Bit-plane error recovery via cross subband for image transmission in JPEG2000", *Multimedia and Expo, 2002. ICME '02. Proceedings. 2002 IEEE International Conference on*, pp. 149.
- Lee, T.S. 1996, "Image representation using 2D Gabor wavelets", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 10, pp. 959-971.
- Li, R., Zeng, B. & Liou, M.L. 1994, "A new three-step search algorithm for block motion estimation", *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 4, no. 4, pp. 438-442.
- Li, X. & Orchard, M.T. 2002, "Novel sequential error-concealment techniques using orientation adaptive interpolation", *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 12, no. 10, pp. 857-864.
- Liu, J., Zhai, G., Yang, X., Yang, B. & Chen, L. 2015, "Spatial Error Concealment With an Adaptive Linear Predictor", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 3, pp. 353-366.
- Liu, L.C. , *A tutorial of wavelet transform*. Available: <http://disp.ee.ntu.edu.tw/tutorial/WaveletTutorial.pdf>.
- Marr, D. & Hildreth, E. 1980, " Theory of Edge Detection ", *Proceedings of the Royal Society of London. Series B, Biological Sciences* , 29 Feb, pp. 187.
- Marvasti, A., Amini, A., Haddadi, F., Soltanolkotabi, M., Khalaj, B., Aldroubi, A., Sanei, S. & Chambers, J. 2012, "A unified approach to sparse signal processing", *EURASIP Journal on Advances in Signal Processing*, vol. 44.
- Meisinger, K. 2007, *Selective Signal Extrapolation and its Application in Image and Video Communications*, Alexander university.
- Meisinger, K. & Kaup, A. 2004, "Spatial error concealment of corrupted image data using frequency selective extrapolation", *Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP '04). IEEE International Conference on*, pp. iii.
- Mohlenkamp, J.M. , *A Tutorial on Wavelets and their Applications* [Homepage of University of Colorado at Boulder Department of Applied Mathematics], [Online]. Available: <http://www.ohio.edu/people/mohlenka/20044/PASIII/waveletIPAM.pdf>.

- N. Feamster and H. Balakrishnan, 2002, "Packet Loss Recovery for Streaming Video," M.I.T. Laboratory for Computer Science, ", *12th International Packet Video Workshop, 2002*, .
- Olivier, R. & Hanqiang, C. 2012, "Nearest Neighbor Value Interpolation", *International Journal of Advanced Computer Science and Applications*, vol. 3, no. 4.
- Padmavathi, S., Priyalakshmi, B. & Soman, K.P. 2012, "Hierarchical Digital Image inpainting Using Wavelets", *Signal & Image Processing: An International Journal (SIPIJ)*, vol. 3, no. 4.
- Pan, J. , *Image Interpolation using Spline Curves* [Homepage of Dept. of Mechanical Engineering, MEC572 term paper, fall, 2003], [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=65A3C5BA3DA45B06691D935C374D342A?doi=10.1.1.517.3701&rep=rep1&type=pdf>.
- Park, J.W., Kim, J.W. & Lee, S.U. 1997, "DCT coefficients recovery-based error concealment technique and its application to the MPEG-2 bit stream error", *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 7, no. 6, pp. 845-854.
- Park, J., Park, D.C., Marks, R.J.,II & El-Sharkawi, M.A. 2005, "Recovery of image blocks using the method of alternating projections", *Image Processing, IEEE Transactions on*, vol. 14, no. 4, pp. 461-474.
- Park, J., Park, D.C., Marks, R.J.,II & El-Sharkawi, M.A. 2002, "Block loss recovery in DCT image encoding using POCS", *Circuits and Systems, 2002. ISCAS 2002. IEEE International Symposium on*, pp. V-245.
- Park, J.W. & Lee, S.U. 1999, "Recovery of corrupted image data based on the NURBS interpolation", *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 9, no. 7, pp. 1003-1008.
- Pauluzzi, D.R. & Beaulieu, N.C. 2000, "A comparison of SNR estimation techniques for the AWGN channel", *Communications, IEEE Transactions on*, vol. 48, no. 10, pp. 1681-1691.
- Petrou, M. & Petrou, C. 2010, *Image Processing the Fundamental*, 2nd edn, Wiley.
- Polikar, R. 2006, Nov 2006-last update, *The Engineer's Ultimate Guide To Wavelet Analysis The Wavelet Tutorial*. Available: <http://users.rowan.edu/~polikar/WAVELETS/WTtutorial.html>.
- Prochazka, A., Vysata, O. & Jerhotova, E. 2010, "Wavelet use for reduction of watershed transform over-segmentation in biomedical images processing", *Information Technology and Applications in Biomedicine (ITAB), 2010 10th IEEE International Conference on*, pp. 1.

- Rabiee, H.R., Radha, H. & Kashyap, R.L. 1996, "Error concealment of still image and video streams with multi-directional recursive nonlinear filters", *Image Processing, 1996. Proceedings., International Conference on*, pp. 37.
- Rane, S.D., Sapiro, G. & Bertalmio, M. 2003, "Structure and texture filling-in of missing image blocks in wireless transmission and compression applications", *Image Processing, IEEE Transactions on*, vol. 12, no. 3, pp. 296-303.
- Rashmi, , Kumar, M. & Saxena, R. 2013, "Algorithm and Technique on Various Edge Detection: A Survey", *Signal & Image Processing: An International Journal (SIPIJ)*, vol. 4, no. 3.
- Rehman, M., Sharif, M. & Raza, M. 2014, "Image Compression: A Survey", *Research Journal of Applied Sciences, Engineering and Technology*, .
- Rombaut, J., Pizurica, A. & Philips, W. 2009, "Passive Error Concealment for Wavelet-Coded I-Frames With an Inhomogeneous Gauss–Markov Random Field Model", *Image Processing, IEEE Transactions on*, vol. 18, no. 4, pp. 783-796.
- Rombaut, J., Pizurica, A. & Philips, W. 2008, "Locally Adaptive Passive Error Concealment for Wavelet Coded Images", *Signal Processing Letters, IEEE*, vol. 15, pp. 178-181.
- Rombaut, J., Pizurica, A. & Philips, W. 2008, "Optimization of Packetization Masks for Image Coding Based on an Objective Cost Function for Desired Packet Spreading", *Image Processing, IEEE Transactions on*, vol. 17, no. 10, pp. 1849-1863.
- Rongfu, Z., Yuanhua, Z. & Xiaodong, H. 2004, "Content-adaptive spatial error concealment for video communication", *Consumer Electronics, IEEE Transactions on*, vol. 50, no. 1, pp. 335-341.
- Salama, P. 1999, *Error Concealment In Encoded Images And Video*, Purdue University.
- Salama, P., Shroff, N. & Delp, E.J. 1997, "A fast suboptimal approach to error concealment in encoded video streams", *Image Processing, 1997. Proceedings., International Conference on*, pp. 101.
- Scho ñlieb, C.-. , *Image Processing – Variational and PDE Methods* [Homepage of Mathematical Tripos Part III], [Online]. Available: http://www.damtp.cam.ac.uk/user/cbs31/Teaching_files/PDEImageLectureNotesLent2014.pdf.
- Seerha, G.K. & Kaur, R. 2013, "Various Edge Detection Methods for Foreground Detection", *International Journal of Computer Science and Mobile Computing*, vol. 2, no. 6, pp. 427-434.

- Seiler, J. & Kaup, A. 2008, "Fast orthogonality deficiency compensation for improved frequency selective image extrapolation", *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, pp. 781.
- Shirani, S., Gallant, M. & Kossentini, F. 2001, "Multiple description image coding using pre- and post-processing", *Information Technology: Coding and Computing, 2001. Proceedings. International Conference on*, pp. 35.
- Shirani, S., Kossentini, F. & Ward, R. 2000, "Reconstruction of baseline JPEG coded images in error prone environments", *Image Processing, IEEE Transactions on*, vol. 9, no. 7, pp. 1292-1299.
- Skodras, A., Christopoulos, C. & Ebrahimi, T. 2001, "The JPEG 2000 still image compression standard", *Signal Processing Magazine, IEEE*, vol. 18, no. 5, pp. 36-58.
- Stigler, S.M. 1986, *The history of statistics*, Stigler, Stephen M.
- Su, D. & Willis, P. 2004, *Image Interpolation by Pixel Level Data-Dependent Triangulation*, Computer graphics forum.
- Suh, J.W. & Ho, Y.S. 1997, "Error concealment based on directional interpolation", *Consumer Electronics, IEEE Transactions on*, vol. 43, no. 3, pp. 295-302.
- Sun, H. & Kwok, W. 1995, "Concealment of damaged block transform coded images using projections onto convex sets", *Image Processing, IEEE Transactions on*, vol. 4, no. 4, pp. 470-477.
- Swati, B., Malviya, N. & Lade, S. 2013, "Analysis of Exemplar Base Inpainting for Adaptive Patch Propagation using Wavelet Transform", *International Journal of Emerging Technology and Advanced Engineering*, vol. 3, no. 5.
- T. Wiegand, G. J. Sullivan, G. Bjontegaard & A. Luthra 2003, "Overview of the H.264/AVC video coding standard", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560-576.
- Tan, K.H. & Ghanbari, M. 1992, "Compact image coding using two-dimensional DCT pyramid", *Electronics Letters*, vol. 28, no. 8, pp. 791-792.
- Teamster, N. & Balakrishnan, H. 2002, "Packet Loss Recovery for Streaming Video", *M.I.T. Laboratory for Computer Science, 12th International Packet Video Workshop*, .
- Thung, K.H. & Raveendran, P. 2009, "A survey of image quality measures", *Technical Postgraduates (TECHPOS), 2009 International Conference for*, pp. 1.

- Tillo, T. & Olmo, G. 2007, "Data-Dependent Pre- and Postprocessing Multiple Description Coding of Images", *Image Processing, IEEE Transactions on*, vol. 16, no. 5, pp. 1269-1280.
- Tschumperle, D. & Deriche, R. 2003, "Vector-valued image regularization with PDE's: a common framework for different applications", *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, pp. I-651.
- Varsa, V., Hannuksela, M.M. & Wang, Y.K. 2001, "Non-normative error concealment algorithms", *ITU-T VCEG-N62*, .
- Walker, J.S. 2008, *A Primer on Wavelets and their Scientific Applications*, 2nd edn, .
- Wang, Y. & Zhu, Q.F. 1998, "Error control and concealment for video communication: a review", *Proceedings of the IEEE*, vol. 86, no. 5, pp. 974-997.
- Wang, Y. & Zhu, Q.F. 1991, "Signal loss recovery in DCT-based image and video codecs", *SPIE 1605, Visual Communications and Image Processing '91: Visual Communication*, pp. 667.
- Wang, Y., Zhu, Q.F. & Shaw, L. 1993, "Maximally smooth image recovery in transform coding", *Communications, IEEE Transactions on*, vol. 41, no. 10, pp. 1544-1551.
- Wang, Z., Bovik, A.C., Sheikh, H.R. & Simoncelli, E.P. 2004, "Image quality assessment: from error visibility to structural similarity", *Image Processing, IEEE Transactions on*, vol. 13, no. 4, pp. 600-612.
- Wang, Z., Yu, Y. & Zhang, D. 1998, "Best neighborhood matching: an information loss restoration technique for block-based image coding systems", *Image Processing, IEEE Transactions on*, vol. 7, no. 7, pp. 1056-1061.
- Wang, Y.K., Hannuksela, M.M., Varsa, V., Hourunranta, A. & Gabbouj, M. 2002, "The error concealment feature in the H.26L test model", *Image Processing. 2002. Proceedings. 2002 International Conference on*, pp. II-729.
- Witkin, A.P. 1984, "Scale-space filtering: A new approach to multi-scale description", *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '84.*, pp. 150.
- Yang, M. & Bourbakis, N.G. 2009, "An Efficient Packet Loss Recovery Methodology for Video Streaming Over IP Networks", *Broadcasting, IEEE Transactions on*, vol. 55, no. 2, pp. 190-201.

- Yaroslavsky, L. , *Digital Image Processing: Applications*. Available:
http://www.eng.tau.ac.il/~yaro/lectnotes/pdf/LI_Images%26ImagingDevices_B.pdf.
- Ye, S., Sun, Q. & Chang, E.C. 2004, "Error concealment for JPEG2000 images based on orthogonal edge directed filters", *Multimedia and Expo, 2004. ICME '04. 2004 IEEE International Conference on*, pp. 1663.
- Zeng, W. & Liu, B. 1995, "Geometric-structure-based directional filtering for error concealment in image/video transmission", *Wireless Data Transmission*, pp. 145.
- Zeng, W. & Liu, B. 1995, "Geometric-structure-based directional filtering for error concealment in image/video transmission", *SPIE 2601, Wireless Data Transmission* SPIE, SPIE Proceedings Vol. 2601: Wireless Data Transmission.
- Zhai, G., Cai, J., Lin, W., Yang, X. & Zhang, W. 2008, "Image error-concealment via Block-based Bilateral Filtering", *Multimedia and Expo, 2008 IEEE International Conference on*, pp. 621.
- Zhai, G., Yang, X., Lin, W. & Zhang, W. 2010, "Bayesian Error Concealment With DCT Pyramid for Images", *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 20, no. 9, pp. 1224-1232.
- Zhang, D. & Wang, Z. may 2002, "Image Information Restoration Based on Long-range Correlation", *ieee transactions on circuits and systems for video technology*, vol. 12, no. 5.
- Zhao, Y., Chen, H., Chi, X. & Jin, J.S. 2005, "Spatial Error Concealment Using Directional Extrapolation", *Digital Image Computing: Techniques and Applications, 2005. DICTA '05. Proceedings 2005*, pp. 41.
- Zhu, Q.F., Wang, Y. & Shaw, L. 1993, "Coding and cell-loss recovery in DCT-based packet video", *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 3, no. 3, pp. 248-258.