AN EBCOT BASED STEREOSCOPIC VIDEO CODEC

by

Nantheera Anantrasirichai

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Engineering

Examination Committee: Dr. Teerapat Sanguankotchakorn (Chairman)
Dr. Kazi M. Ahmed
Dr. Nitin Kumar Tripathi

Nationality: Thai
Previous Degree: Bachelor of Engineering in Electrical Engineering
Chulalongkorn University
Bangkok, Thailand

Scholarship Donor: His Majesty the King of Thailand

Asian Institute of Technology
School of Advanced Technologies
Thailand
December 2003
ACKNOWLEDGEMENTS

I would like to state my sincere gratitude to Dr. Teerapat Sanguankotchakorn, my thesis advisor, for his valuable advice, excellent guidance and constant encouragement throughout my thesis period. Without his contributive help, this thesis would not have been completed successfully.

My honest thanks should also go to Dr. Kazi M. Ahmed and Dr. Nitin Kumar Tripathi for their kind consideration to serve as my examination committee members, and their valuable comments and suggestions to enhance this work.

I also would like to give the profound thank to Dr. W.A.C. Fernando at Brunel University and Dr. Eran Edirisinghe at Loughborough University for their munificent assistance and helpful discussion.

I would like to extend my thanks to all faculty members for imparting invaluable knowledge, and also for my seniors, juniors, all Lap supervisors, staffs and friends for their warm friendship, kindness and encouragement. My sincere appreciation is comprehensive to Ms. Suthinee Thanapirom for her helpful elementary knowledge. Thanks are also due to the secretaries of Telecommunication Program for their kindness and useful help with smiles.

I had been indebted to my donor, His Majesty the King of Thailand, for the full financial support. Without this support, I would not have been able to profoundly understand such invaluable knowledge gained from all people in AIT.

Finally, all my deep and sincere thanks have been expressed to my parents and family members as well as my close friends for their love, moral support and constant encouragement.
ABSTRACT

Over the past decade, the realistic visual communication systems have been dramatically grown and have been useful for various purposes. So far, however, the amount of information of a three-dimensional imaging system has limited many commercial applications, since this system results in a two fold increase in bandwidth over the existing monoscopic channel bandwidth. Thus, the efficient compression algorithms are vital to reduce the size of data without sacrificing the perceived quality.

This thesis has been set to develop a framework for stereo video compression by exploiting spatial correlation, temporal correlation and high correlation between stereo image pair in order to increase the quality of stereo video at the specific bit rate with the Embedded Block Coding with Optimized Truncation (EBCOT) coding. The proposed stereo video codec is composed of two sub-codecs for coding and decoding two image streams, namely the main stream that independently coded by an EBCOT video codec and, secondly, the auxiliary stream that is predicted based on disparity compensation and coded the residual frames.

The experimental results of comparing the performance of the proposed codec with the DCT based stereo video codec and ZTE based stereo video codec show outperforms in coding both main and auxiliary streams. Apart from the significant PSNR gain, proposed codec has various other advantages such as easier to control the desired size, SNR scalability, resolution scalability, smaller buffer need, random access capability and good error resilience. Therefore, the EBCOT based stereoscopic video codec is highly attractive for the next generation of multimedia communications.
## Table of Contents

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Title Page</td>
<td>i</td>
</tr>
<tr>
<td></td>
<td>Acknowledgements</td>
<td>ii</td>
</tr>
<tr>
<td></td>
<td>Abstract</td>
<td>iii</td>
</tr>
<tr>
<td></td>
<td>Table of Contents</td>
<td>iv</td>
</tr>
<tr>
<td></td>
<td>List of Abbreviations</td>
<td>vi</td>
</tr>
<tr>
<td></td>
<td>List of Symbols</td>
<td>vii</td>
</tr>
<tr>
<td></td>
<td>List of Figures</td>
<td>x</td>
</tr>
<tr>
<td></td>
<td>List of Tables</td>
<td>xii</td>
</tr>
<tr>
<td>1</td>
<td>Introduction</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1.1 Background</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1.2 Statement of Problem</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>1.3 Objectives</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>1.4 Limitations</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>Literature Review</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>2.1 Stereo Video</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>2.1.1 Correlations</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>2.1.2 Occlusions</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>2.2 Video Coding in MPEG-2 Standard</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>2.2.1 Temporal Picture Structure</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>2.2.2 Stereoscopic Coding</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>2.3 Motion and Disparity Estimation</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>2.4 Overlapping Block Motion/Disparity Compensation</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>2.5 Embedded Block Coding with Optimized Truncation (EBCOT)</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>2.5.1 Discrete Wavelet Transform (DWT)</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>2.5.2 Quantization/Dequantization</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>2.5.3 Tier-1 Encoder</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>2.5.4 Tier-2 Encoder</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>2.5.5 Rate Control</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td>2.6 Picture Quality Assessment</td>
<td>14</td>
</tr>
<tr>
<td>3</td>
<td>Methodology</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>3.1 Proposed System Model</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>3.1.1 Main Stream Encoder</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>3.1.2 Auxiliary Stream Encoder</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td>3.2 System Specifications</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>3.2.1 Motion Estimation</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>3.2.2 Overlapped Block Motion Compensation</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>3.2.3 Disparity Estimation and Compensation</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>3.2.4 EBCOT Encoder</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>3.2.5 Rate Control</td>
<td>21</td>
</tr>
<tr>
<td></td>
<td>3.3 Enhancement Techniques for Stereoscopic Compression</td>
<td>21</td>
</tr>
<tr>
<td>Chapter</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>---------</td>
<td>-----------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>4</td>
<td>Results and Discussions</td>
<td>22</td>
</tr>
<tr>
<td></td>
<td>4.1 Results for the Gray Scale Stereo Image Pairs</td>
<td>22</td>
</tr>
<tr>
<td></td>
<td>4.2 Results for the Colour Stereo Image Pairs</td>
<td>26</td>
</tr>
<tr>
<td></td>
<td>4.3 Results for Concerning the Occlusion in Stereo Image Pair</td>
<td>28</td>
</tr>
<tr>
<td></td>
<td>4.4 Results for Stereo Video Sequences</td>
<td>29</td>
</tr>
<tr>
<td></td>
<td>4.5 Results of the Effect of the Reference View Selection</td>
<td>37</td>
</tr>
<tr>
<td>5</td>
<td>Conclusions and Recommendations</td>
<td>39</td>
</tr>
<tr>
<td></td>
<td>5.1 Conclusions</td>
<td>39</td>
</tr>
<tr>
<td></td>
<td>5.2 Recommendations</td>
<td>40</td>
</tr>
<tr>
<td></td>
<td>References</td>
<td>41</td>
</tr>
</tbody>
</table>

Appendix A : Stereoscopic Video Codec Analysis 45
Appendix B : Variable Scanning Range for Disparity Estimation 61
Appendix C : Occluded Prediction Analysis 65
List of Abbreviations

2D Two-Dimensional
3D Three-Dimensional
3G Third Generation wireless system
B-frame Bidirectionally predictive coded frame
BMA Block Matching Algorithms
bpp Bit per pixel
CAD Computer Aid Design
DCD Disparity Compensated Difference
DCP Disparity Compensated Prediction
DCT Discrete Cosine Transform
DFD Displaced Frame Difference
DV Disparity Vector
DWT Discrete Wavelet Transform
EBCOT Embedded Block Coding with Optimized Truncation
EZW Embedded Zerotree Wavelet
FSBM Fixed-Size Block Matching
GOP Group of Pictures
HDTV High Definition Television
ICT Irreversible Colour Transform
I-frame Intra coded frame
ISO International Standards Organization
ITU International Telecommunication Union
ITU-T International Telecommunication Union-Telecommunication
JPEG Joint Photographic Experts Group
MAE Mean Absolute Error
MCP Motion Compensated Prediction
ME Motion Estimator
MPEG Motion Picture Experts Group
MR Magnitude Refinement
MSE Mean Squared Error
MV Motion Vector
OBDC Overlapped block disparity compensation
OBMC Overlapped block motion compensation
PCRD Post-Compression Rate-Distortion
P-frame Predictive coded frame
PSNR Peak-to-Peak Signal to Noise Ratio
QMF Quadrature Mirror Filters
RCT Reversible Colour Transform
RGB Red-Green-Blue (Color model)
RLC Run Length Coding
SAD Sum of Absolute Difference
SC Sign Coding
SPIHT Set Partitioning In hierarchical Trees
VLC Variable Length Coding
VR Virtual Reality
ZC Zero Coding
ZTE Zerotree Entropy Coding
### List of Symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\sigma(m,n)$</td>
<td>Second state variable</td>
</tr>
<tr>
<td>$\delta_z$</td>
<td>Overhead of disparity vectors in variable scanning range scheme</td>
</tr>
<tr>
<td>$\lambda$</td>
<td>Real number.</td>
</tr>
<tr>
<td>$\eta$</td>
<td>The number of scanning range</td>
</tr>
<tr>
<td>$\sigma(\cdot)$</td>
<td>The significance of pixel</td>
</tr>
<tr>
<td>$\mu_b$</td>
<td>Mantissa of sub-band, $b$</td>
</tr>
<tr>
<td>$\Delta_b$</td>
<td>Quantizer Step Size of sub-band, $b$</td>
</tr>
<tr>
<td>$\epsilon_b$</td>
<td>Exponent of sub-band, $b$</td>
</tr>
<tr>
<td>$B$</td>
<td>Baseline distance</td>
</tr>
<tr>
<td>$B_i$</td>
<td>Code-blocks</td>
</tr>
<tr>
<td>$B_{ij}$</td>
<td>Indicated Blue dominance of pixel at row $i^{th}$ and column $j^{th}$</td>
</tr>
<tr>
<td>$B_{kl}$</td>
<td>Block for disparity/motion estimation</td>
</tr>
<tr>
<td>$b_i$</td>
<td>Sub-band</td>
</tr>
<tr>
<td>$b_p$</td>
<td>Bit pixel sample values</td>
</tr>
<tr>
<td>$\hat{b}_{\text{dis}}^{\text{uni}}$</td>
<td>The number of bits used for disparity vector before coding for uniform scanning range</td>
</tr>
<tr>
<td>$\hat{b}_{\text{dis}}^{\text{var}}$</td>
<td>The number of bits used for disparity vector before coding for variable scanning range</td>
</tr>
<tr>
<td>$\hat{b}_{\text{mot}}$</td>
<td>The number of bits used for motion vector before coding</td>
</tr>
<tr>
<td>$C_1$</td>
<td>Left coordinate</td>
</tr>
<tr>
<td>$C_r$</td>
<td>Right coordinate</td>
</tr>
<tr>
<td>$C_w$</td>
<td>World coordinate</td>
</tr>
<tr>
<td>$D$</td>
<td>Overall distortion</td>
</tr>
<tr>
<td>$D_i$</td>
<td>Distortion</td>
</tr>
<tr>
<td>$\hat{D}$</td>
<td>Distortion Matrix</td>
</tr>
<tr>
<td>$F$</td>
<td>Focal length of camera</td>
</tr>
<tr>
<td>$G[n]$</td>
<td>High-pass filter</td>
</tr>
<tr>
<td>$G_{ij}$</td>
<td>Indicated Green dominance of pixel at row $i^{th}$ and column $j^{th}$</td>
</tr>
<tr>
<td>$H[n]$</td>
<td>Low-pass filter</td>
</tr>
<tr>
<td>$HH_L$</td>
<td>High-pass sub-band at resolution level, $L$</td>
</tr>
<tr>
<td>$h_i(k)$</td>
<td>Horizontal contribution</td>
</tr>
<tr>
<td>$HL_L$</td>
<td>Horizontally high-pass at resolution level, $L$</td>
</tr>
<tr>
<td>$h_{\text{mot}}$</td>
<td>Height of the variable scanning area in motion estimation</td>
</tr>
</tbody>
</table>
Height of the uniform scanning area in disparity estimation

Height of the variable scanning area in disparity estimation

Position of pixel in row

Position of pixel in column

Position on horizontal axis

Position on vertical axis

Vertically high-pass at resolution level, L

Successive resolution level

Low-pass sub-band at resolution level, L

Height of the reference frame and the predicted frame

Height of the block for disparity/motion estimation

Height of the sub-block for defining scanning range

Maximum

Minimum

Group of Pictures length

Width of the reference frame and the predicted frame

Width of the block for disparity/motion estimation

Width of the sub-block for defining scanning range

Subset of feasible truncation points

Truncation point

Set of truncation points

Overhead bit for identify occlusion

The number of bits of overhead z used in disparity estimation

Values of pixel at row $i^{th}$ and column $j^{th}$ in Y, U and V component, respectively, of the predicted frame

Values of pixel at row $i^{th}$ and column $j^{th}$ in Red, Green and Blue component, respectively, of the predicted frame

Values of pixel at row $i^{th}$ and column $j^{th}$ in Y, U and V component, respectively, of the reference frame

Values of pixel at row $i^{th}$ and column $j^{th}$ in Red, Green and Blue component, respectively, of the reference frame

Occlusion block with top-left pixel at row $i^{th}$ and column $j^{th}$

Values of pixel at row $i^{th}$ and column $j^{th}$ of Residual frame

Values of pixel at row $i^{th}$ and column $j^{th}$ of reconstructed Residual frame

Values of pixel at row $i^{th}$ and column $j^{th}$ of interleaved frame, Residual and Occlusion

Values of pixel at row $i^{th}$ and column $j^{th}$ in luminance component of Residual frame
\( R_{ij} \) Indicated Red dominance of pixel at row \( i^{th} \) and column \( j^{th} \)  
\( r \) Bias Parameter  
\( S_{i}^{j} \) distortion-rate slope  
\( s_{i}[k] \) 2D sequences of sub-band samples in code-block \( B_{i} \)  
\( \hat{s}_{i}^{n}[k] \) Quantized representation of samples associated with truncation point \( n_{i} \)  
\( U_{ij} \) Indicated Undefined colour dominance of pixel at row \( i^{th} \) and column \( j^{th} \)  
\( U(x; y) \) Quantizer Input  
\( V(x; y) \) Quantizer Output  
\( v_{i}(k) \) Vertical contribution  
\( W \) Weighting coefficient of Overlapped Block Motion Compensation  
\( w_{mot} \) Width of the variable scanning area in motion estimation  
\( w_{uni} \) Width of the uniform scanning area in disparity estimation  
\( w_{var} \) Width of the variable scanning area in disparity estimation  
\( w_{b_{i}} \) Coefficient weighting for the sub-band, \( b_{i} \)  
\( x_{i} \) Values of a pixel of an image  
\( x_{l} \) Left image coordinate  
\( x_{r} \) Right image coordinate  
\( X_{pre} \) Predicted value of the Overlapped Block Motion Compensation  
\( y_{i} \) Values of a pixel of the reconstructed image
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>A 3D perception</td>
</tr>
<tr>
<td>2.2</td>
<td>Example of temporal picture structure</td>
</tr>
<tr>
<td>2.3</td>
<td>A structure of stereoscopic codec derived from temporal scalability</td>
</tr>
<tr>
<td>2.4</td>
<td>MPEG-2 Multiview profile</td>
</tr>
<tr>
<td>2.5</td>
<td>Stereo disparity</td>
</tr>
<tr>
<td>2.6</td>
<td>Overlapping block motion compensation</td>
</tr>
<tr>
<td>2.7</td>
<td>The structure of codec (a) Encoder and (b) Decoder</td>
</tr>
<tr>
<td>2.8</td>
<td>Wavelet decomposition structure for three decomposition levels</td>
</tr>
<tr>
<td>2.9</td>
<td>The parent-child relationship of wavelet coefficients</td>
</tr>
<tr>
<td>2.10</td>
<td>Two-tiered coding structure</td>
</tr>
<tr>
<td>2.11</td>
<td>Illustration of block aid to bit-stream layers</td>
</tr>
<tr>
<td>3.1</td>
<td>The proposed stereo video encoder</td>
</tr>
<tr>
<td>3.2</td>
<td>Prediction process in stereo encoder</td>
</tr>
<tr>
<td>3.3</td>
<td>Main Stream Encoder</td>
</tr>
<tr>
<td>3.4</td>
<td>Auxiliary Stream Encoder</td>
</tr>
<tr>
<td>3.5</td>
<td>Disparity/motion estimation and compensation in the auxiliary stream encoder</td>
</tr>
<tr>
<td>3.6</td>
<td>Aqua stereo image pair (a) Right Image (b) Prediction Error Image</td>
</tr>
<tr>
<td>3.7</td>
<td>The uniform scalar quantization with dead zone</td>
</tr>
<tr>
<td>4.1</td>
<td>The rate-distortion performance of reconstructed stereo images. (a) Aqua (b) Tunnel (c) Manege</td>
</tr>
<tr>
<td>4.2</td>
<td>The right images of ‘Tunnel’ coded at 0.25 bpp for image pair. The original right image illustrated in (a) and its reconstructed images from (b) DCT coding (c) ZTE coding (D) EBCOT coding</td>
</tr>
<tr>
<td>4.3</td>
<td>The rate-distortion performance of reconstructed right images illustrated in RGB colour domain. (a) Aqua (b) Tunnel (c) Manege</td>
</tr>
<tr>
<td>4.4</td>
<td>The right images of ‘Manege’ coded at 0.25 bpp for image pair. The original right image illustrated in (a) and its reconstructed images from (b) Uniform scanning range (c) Variable scanning range</td>
</tr>
<tr>
<td>4.5</td>
<td>Improvement of the proposed scheme over the conventional scheme applied to (a) Claude2 (b) Claude4 (c) Booksale (frame 1) (d) Crowd (frame 1)</td>
</tr>
<tr>
<td>4.6</td>
<td>The subjective results in frame 30 of Booksale sequences (AVDS Sequences) coded at 0.244 bpp. (a) The cropped original right image (b) The cropped right image of DCT coding (c) The cropped right image of ZTE coding (d) The cropped right image of EBCOT coding</td>
</tr>
<tr>
<td>4.7</td>
<td>The subjective results in frame 63 of Crowd sequences (AVDS Sequences) coded at 0.472 bpp (a) The cropped original right image (b) The cropped right image of DCT coding (c) The cropped right image of ZTE coding (d) The cropped right image of EBCOT coding</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>--------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>A.1</td>
<td>Frame size $M \times N$ with blocks size $m \times n$</td>
</tr>
<tr>
<td>A.2</td>
<td>Interpolation of the half-pixel refinement</td>
</tr>
<tr>
<td>A.3</td>
<td>Sub-block for overlapped block motion/disparity compensation</td>
</tr>
<tr>
<td>A.4</td>
<td>Sample scan order within a code-block</td>
</tr>
<tr>
<td>A.5</td>
<td>Neighbors of current bit</td>
</tr>
<tr>
<td>B.1</td>
<td>Comparing components of the Baboon image. (a) Original image. (b) Luminance component or Y component. (c) Red component. (d) Green component and (e) Blue component</td>
</tr>
<tr>
<td>B.2</td>
<td>Obtaining the blue dominant object by removing unmarked pixels and pixels marked with other colours: (a) Original image (b) Identified blue dominant object</td>
</tr>
<tr>
<td>B.3</td>
<td>Correspondence between blue dominant components of the stereo image pair (a) left image object (b) right image object</td>
</tr>
<tr>
<td>C.1</td>
<td>Reconstructed Left image of Claude4-3 with 1.44 bpp (PSNR = 40.94 dB)</td>
</tr>
<tr>
<td>C.2</td>
<td>The original Right image of Claude4-4 that would be coded with 1.2 bpp</td>
</tr>
<tr>
<td>C.3</td>
<td>Detected edges of the Right image used Canny method with threshold 0.3 for weak edges and 4 for strong edges</td>
</tr>
<tr>
<td>C.4</td>
<td>Estimated occluded region in the Right image by Edge detection method (Block size = 16x16, $r = 3$)</td>
</tr>
<tr>
<td>C.5</td>
<td>Parallel camera configuration a) 3D view b) the X-Z plane view</td>
</tr>
<tr>
<td>C.6</td>
<td>The shade blocks identify the predicting occluded region</td>
</tr>
<tr>
<td>C.7</td>
<td>The final occluded estimation prepared appropriately for EBCOT algorithm</td>
</tr>
<tr>
<td>C.8</td>
<td>Codec for occlusion consideration (a) encoder (b) decoder</td>
</tr>
</tbody>
</table>
### List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Daubechies 9/7 analysis and synthesis filter coefficients</td>
<td>20</td>
</tr>
<tr>
<td>4.1</td>
<td>The average PSNRs of <em>Booksale</em> sequences for the different type of pictures at 0.244 bpp</td>
<td>30</td>
</tr>
<tr>
<td>4.2</td>
<td>The average PSNRs of <em>Crowd</em> sequences for the different type of pictures at 0.244 bpp</td>
<td>30</td>
</tr>
<tr>
<td>4.3</td>
<td>The average PSNRs of <em>Booksale</em> sequences for the different type of pictures at 0.34 bpp</td>
<td>31</td>
</tr>
<tr>
<td>4.4</td>
<td>The average PSNRs of <em>Crowd</em> sequences for the different type of pictures at 0.34 bpp</td>
<td>31</td>
</tr>
<tr>
<td>4.5</td>
<td>The average PSNRs of <em>Booksale</em> sequences for the different type of pictures at 0.472 bpp</td>
<td>32</td>
</tr>
<tr>
<td>4.6</td>
<td>The average PSNRs of <em>Crowd</em> sequences for the different type of pictures at 0.472 bpp</td>
<td>32</td>
</tr>
<tr>
<td>4.7</td>
<td>The average PSNRs of <em>Booksale</em> sequences for the different type of pictures at 1.00 bpp</td>
<td>33</td>
</tr>
<tr>
<td>4.8</td>
<td>The average PSNRs of <em>Crowd</em> sequences for the different type of pictures at 1.00 bpp</td>
<td>33</td>
</tr>
<tr>
<td>4.9</td>
<td>The average PSNRs of <em>Booksale</em> sequences comparing between using Left view and Right view as the main stream at 0.30 bpp</td>
<td>37</td>
</tr>
<tr>
<td>4.10</td>
<td>The average PSNRs of <em>Crowd</em> sequences comparing between using Left view and Right view as the main stream at 0.50 bpp</td>
<td>37</td>
</tr>
<tr>
<td>A.1</td>
<td>The code of position of first bit in a stripe</td>
<td>55</td>
</tr>
<tr>
<td>A.2</td>
<td>Assignment of nine ZC contexts</td>
<td>56</td>
</tr>
<tr>
<td>A.3</td>
<td>Assignment of three MR contexts</td>
<td>56</td>
</tr>
<tr>
<td>A.4</td>
<td>Assignment of three SC contexts</td>
<td>57</td>
</tr>
</tbody>
</table>
CHAPTER 1
INTRODUCTION

1.1 Background

In recent times, as the result of having expeditiously developed of the two-dimensional (2D) visual communication technologies, various commercial systems are available for real-time visual communication based on standards established for different application such as JPEG (Gregory, 1991), MPEG-1 (MPEG-1, 1991), MPEG-2 (MPEG-2, 1994) and MPEG-4 (Sikora, 1997), H.261 (ITU-T, 1993) or H.263 (ITU-T, 1998). Nevertheless, they might not be sufficient for the increasing new demands for realism or more natural representations of the scene. To achieve these requirements, the innovative systems may have to assimilate various human senses, namely sight, hearing, touch, smell, and taste. By including these senses, the imaging related technologies have been developed. For example, the first stage of black and white TV was designed to extend the sense of sight and hearing and then colours were integrated to represent realistic scene. Recent activities on HDTV are another effort to add realism in the 2D display.

The new development of auto stereoscopic monitor technology has been opening a new way for the displaying images and video sequences in three-dimensional (3D) without the aid of special eyeglasses or helmet-mounted display kits that frequently result in noticeable user discomfort. Therefore, the 3D TV without special glasses will be the driving step of the TV systems and numerous applications. Eventually, various technologies such as 3D TV, telephone, computer, and consumer virtual reality (VR) technologies are expected to congregate and offer the combined interactive functionality to provide the virtual reality. The stereo (or multi-view) images are bound to have a number of main applications in near future in the fields of 3D visualization (CAD), 3D telemedicine, 3D telerobotics, 3D visual communications, 3D TV and cinema, and Virtual Reality.

To motivate our sense of stereovision, two pictures of the same scene from two horizontally separated positions presented the left frame to the left eye and the right frame to the right eye are required. The human brain can process the difference between these two images to make out the depth information of the scene being viewed that yield 3D perception. Hence, two 2D image frames can represent a 3D image. These frames are said to form a stereo image pair. When a novel stereo pair is stored and transmitted, twice as many bits will be required to represent stereo pair compared to a single image and make the result that increasing unacceptable and impractical when transmitting stereo over limited channel bandwidth. This is an elementary problem of data compression.

Fortunately, the left and right images of a stereo pair have mostly correlation; the compression can be developed continually over the last decade. Many stereo image compression algorithms have been studied based on intensity or features. The first technique is to code sum and difference of the two images in a stereo pair (Perkins, 1992). Afterward 3D discrete cosine transform (DCT) coding of stereo image was presented (Dinstein et al., 1988). It is equivalent to the sum-difference coding in transform domain. The performance of these two techniques decreases with increased disparity values by assuming that objects in the scene have same disparity values. Nevertheless, these methods
are not principally efficient since objects in the scene have normally different disparity values. Hence, the concept of disparity compensation, which established correspondence between similar areas in a stereo pair using binocular disparity information, was constituted (Lukacs, 1986). This method is used to predict the rest of the views from an independently coded view. Other disparity compensation based methods have also been proposed. The disparity compensation based coding as a conditional coding approach that is optimal for lossless coding and suboptimal for lossy coding has been formalized (Perkins, 1992). The results of MPEG-2 compatible coding have been presented (Puri et al., 1995). In the standard, one view is coded as the base layer and another view is coded within enhancement layer of the temporal scalability model of the MPEG-2 standard.

According to having a lot of redundancy of the stereo video, the two image streams could be compressed and transmitted through a single monocular channel’s bandwidth without excessively sacrificing the perceived stereoscopic image quality. The stereoscopic image sequences are compressed as still stereo image pair that exploits the high spatial correlation between the left and right images, while the temporal correlation between frames can be taken along the lines of the MPEG standards to achieve further compression. Auspiciously, in human vision systems, one eye is dominant over the other eye; the dominant eye will mask artifacts viewed by the non-dominant eye. This feature allows an unequal quality of image between the two views.

Moreover, the coding algorithms are purposed to achieve efficiently image compression as well as spatial SNR scalability. In video coding, firstly, the coding principle is based on DCT; however, according to appearance of blocking artifacts and mosquito noises of DCT, the state of wavelet based coding is more attractive and has solved these problems significantly. Because of this advantage, the discrete wavelet transform (DWT) can be broaden to the stereoscopic video coding as well. With wavelet transform algorithm, various embedded coding structures are intended. For instance, the zero-tree structure combined with bit plane coding is an efficient and very simple compression scheme for the DWT (Shapiro, 1993). The embedded zero-tree wavelet (EZW) coding scheme has proven its efficiency and flexibility in still image coding in terms of image quality and computational simplicity. Following the introduction of the zerotree prediction methodology by Shapiro in EZW, a variant of coding of wavelet coefficients by successive approximation have been presented (Said and Pearlman, 1996), that even without arithmetic coding outperforms EZW, called spatial partitioning of images into hierarchical trees (SPIHT).

The other new fascinating image coding methods are Zerotree Entropy Coding (ZTE) and Embedded Block Coding with Optimized Truncation (EBCOT). Zerotree Entropy Coding (ZTE) is first proposed (Martucci and Sodagar, 1996) and further modified video coder exploiting ZTE is published (Martucci et al., 1997). ZTE was developed at the David Sarnoff Research Center and submitted to MPEG-4 video coder based on wavelets to deliver both high compression and to enhance scalability functionalities. For EBCOT algorithm (Taubman, 2000), it is a scalable compression method like EZW and SPIHT. This algorithm uses the concept of fractional bitplane algorithm, arithmetic coder and rate distortion optimization instead of zerotree technique. The idea of this algorithm is the separation of the bits in transformed image components into various significant groups which are compressed by using arithmetic coder. Then, the least compressed significant groups are truncated until it achieves the desired size of final bit stream. The main advantages of EBCOT algorithm are easier to control the desired size than the older algorithms, progressive recovery of an image by fidelity or resolution,
region of interest coding, smaller buffer need, random access to particular regions of an image without needing to decode the entire bit stream, a flexible file format and good error resilience. By contrast, EZW and SPIHT offer only SNR scalability; the EBCOT algorithm offers SNR scalability and resolution scalability concurrently. All these features have made EBCOT to be the chosen coding method for the JPEG2000 standard. Not only for still images but it is also used for image sequences, called Motion JPEG2000, which leverages the powerful wavelet based JPEG2000 Part I standard. Hence, EBCOT may be charismatic for stereo video coding as well.

1.2 Statement of Problem

According to the limitation of number of frequencies used in channel, the transmission of double bandwidth of stereo video sequences compared with the conventional 2D system is impracticable. Moreover, twice-needed buffer for storing enormous amount of data is required. Consequently, the appropriate efficient compression algorithms are imperative to reduce the size of data requirement while sustain images’ quality after decoding. In addition, high quality and optional functionalities, such as inclusive scalabilities, of stereo video compression are continually enhanced in order to support various requirements in the next generation of multimedia.

1.3 Objectives

The major purpose of this thesis is set to develop a framework for stereo video compression by exploiting spatial correlation, temporal correlation and high correlation between stereo image pair in order to increase the quality of stereo video at a specific bit rate with the EBCOT coding.

Specific objectives of this work can be listed as follows.

- Study the stereo video coding concept, EBCOT coding and related works.
- Develop an algorithm for stereo image coding and stereo video coding with the EBCOT.
- Investigate the performance of the proposed schemes.
- Compare the performance of the proposed schemes with DCT based stereo video codecs and ZTE based codecs.

1.4 Limitations

Assume that the sequences used in the simulation are from parallel camera configurations so that the displacement between the left and right image coordinates of the same 3D position known as disparity has only a horizontal component and is related to the depth of that 3D point.
CHAPTER 2
LITERATURE REVIEW

This chapter provides the comprehensive literature review relating to the concept of stereo video, research work done in the area of Motion and Disparity estimation, Overlapping Block Motion/Disparity Compensation and the fundamental background of Embedded Block Coding with Optimized Truncation (EBCOT).

2.1 Stereo Video

The stereo video performs 3D for video sequences by using a stereovision technique for providing depth cues and displays two realism 2D sequences simultaneously to obtain the depth perception. In human visual system, based on a process named stereopsis, a 3D stereo picture results from the combination of the two images received by the brain from each eye that views the world from a slightly different vantage point as shown in Figure 2.1

![Figure 2.1: A 3D perception (Copper, 2001)](image)

The fusion of these two slightly different pictures gives the sensation of strong three-dimensionality or relative depth by matching up the similarities and adding in the small differences. At near, there is a greater difference in what the two eyes view as compared to far. Thus, stereopsis is strongest and most important at near distances.

2.1.1 Correlations

Siegel et al. (1994) have identified four kinds of correlations or redundancies that can be exploited to compress stereoscopic images.

i) **Spatial correlation.** Within a single frame, large areas with little variation in intensity and colour permit efficient encoding based on internal predictability, i.e., the fact that any given pixel is most likely to be identical or nearly identical to its neighbors. This is the basis for most conventional still image compression methods.

ii) **Temporal correlation.** Between frames in a motion sequence, large areas in rigid-body motion permit efficient coding based on frame-to-frame predictability. The approach is fundamentally to transmit an occasional frame, and interpolate coefficients that
permit the receiver to synthesize reasonable approximations to the intermediate frames.

iii) **Perspective correlation.** Between frames in a binocular stereoscopic image pair, large areas differing only by small horizontal offsets permit efficient coding based on disparity predictability.

iv) **WorldLine correlation.** The WorldLine concept refers to the event when vertical and axial velocities are small and horizontal motion suitably compensates perspective; time-offset frames in the left and right image streams can be nearly identical. WorldLine correlation is the combination of temporal correlation and perspective correlation; the most interesting manifestation of WorldLine correlation is the potential near-identity of appropriately time-offset frames in the left and right image streams respectively. The concept is useful for situations in which the camera is fixed and parts of the scene are in motion, the scene is fixed and the camera is in motion, and both the camera and parts of the scene are in motion.

The first two are conventional image compression methods that might be applied to two stereoscopic views independently. The third kind applies to still image pairs, or temporally corresponding members of a motion stream pair. The fourth kind, which is really a combination of the second and third kinds, is applied to motion stream stereo pairs.

### 2.1.2 Occlusions

Although, a stereo pair has much correlation or redundancy, one obstacle of the 3D image compression is the difficulty to cope with occlusions, i.e., features that can be seen from only one of the two perspectives. These may cause hard or erroneous matching between the left and right images. There are two kinds of occlusion that may arise in stereoscopic image ([Frajka and Zeger](#), 2002).

i) Finite viewing area occurs on the left side of the left image and the right side of the right image where the respective eye can see objects that the other eye cannot.

ii) Depth discontinuity is due to overlapping objects in the image; certain portion can be converted from one eye on which the other eye has direct sight.

The occlusion that is the result of foreground objects blocking visibility of background objects is inevitable (except perhaps in some highly artificial laboratory scenario) and must be resolved by potentially expensive and potentially brittle ad hoc heuristic detection.

### 2.2 Video Coding in MPEG-2 Standard

#### 2.2.1 Temporal Picture Structure

Three main picture types are defined. Firstly, Intra coded pictures (I-Pictures) are coded without reference to other pictures. They provide access points to the coded sequences where decoding can begin, but are coded with only moderate compression. Secondly, Predictive coded pictures (P-Pictures) are coded more efficiently using motion compensated prediction from a past intra or predictive coded picture and are generally used as a reference for further prediction. Finally, Bidirectionally-predictive coded pictures (B-Pictures) provide the highest degree of compression but require both past and future
reference pictures for motion compensation. The organization of the three picture types in a sequence is very flexible. The choice is left to the encoder and will depend on the requirements of the application. Figure 2.2 illustrates an example of the relationship among the three different picture types.

![Temporal Picture Structure](image)

**Figure 2.2: Example of temporal picture structure (ITU-T, 1995)**

A Group of pictures (GOP) is a series of one or more pictures to assist the random access into the picture sequences. The first coded picture in the GOP is an I-picture. It is followed by a sequence of P- and B-pictures arranged in a particular order.

The GOP length, usually represented by \( N \), is defined as the distance between I-pictures. The group of pictures may be of any length, but it should be at least one I-picture in each GOP. Applications requiring random access, fast forward play or fast and normal reverse play may use short GOPs. GOP may also start at scene cuts or other cases where motion compensation is not effective. The number of consecutive B-pictures is variable. Neither a P- nor a B-picture needs to be presented. Actual values of \( N \) are not defined in the standard. This reordering introduces delays of equal to the number of B-pictures between the reference I- and P-pictures at the encoder. The same amount of delay is introduced at the decoder by putting the decoding sequences back to its original.

### 2.2.2 Stereoscopic Coding

The extension of the MPEG-2 video standard for multi-view applications (e.g. used for stereoscopic video) has been promoted to a final International Standard. It foresees higher compression of the right view of stereoscopic video by exploiting the similarity between the left and the right view (ITU-T, 1996).

As the temporal scalability in MPEG-2 refers to layered coding that produces two or more layers. It is possible to encode a base layer stream representing a signal with a reduced frame rate, and then defines an enhancement layer stream (or auxiliary layer), which can be used to insert additional frames in between to allow reproduction with full frame rate if both streams are available. Frames from one camera view (usually the left) are defined as the base layer, and frames from the other one(s) as enhancement layer(s). It is suggested to code two stereoscopic video streams. The operation of the codec is shown in Figure 2.3.

The left view is applied to a base view encoder and the right view (or called enhancement layer) is applied to the temporal auxiliary view encoder, an MPEG-2
temporal interlayer encoder that uses temporal prediction from the decoded base layer, the decoded left view. Two-output bit streams are packetized by system multiplexer, for transmission or storage. At the decoder, after unpacketized, the bitstreams are separated to its layers and fed to base view decoder and temporal auxiliary view decoder which are the no scalable MPEG-2 video decoder and MPEG-2 temporal interlayer decoder respectively.

Figure 2.3: A structure of stereoscopic codec derived from temporal scalability (Haskell et al., 1996)

The enhancement layer can be coded with MPEG-2 temporal interlayer coding with predictions with respect to decoded left-view frames. An image is coded in either the P-picture or B-picture. The P-picture is predictively coded using the corresponding frame in the left reference. The B-picture can reference from a previous frame in the right view and the corresponding left view frame. The motion vector used in the latter case is actually a disparity vector, which is used in prediction disparity compensated prediction process. This encoding scheme is shown in the Figure 2.4.
prediction may be based on disparity compensation or the best of motion and disparity compensation.

While the motion prediction estimates the moving object between two frame sequences, disparity performs the displacements between the 2D points in the left and the right image that correspond to the same 3D point in a real scene. In the other words, if left-right pairs are selected from 3D-stereoscopic motion streams at different times, such that perspective-induced disparity left-right and motion-induced disparity earlier-later produce about the same visual effect, then extremely high correlation will exist between the members of these pairs.

Considering Figure 2.5, given a point A in the left picture, its matching point B in the right picture does not in general lie directly underneath A. The vector connecting B to A has called disparity.

2.4 Overlapping Block Motion/Disparity Compensation

In stereo video coding, a motion and disparity estimated approach exploited the similarity find a motion/disparity vector field for one of the images and then transmit the disparity compensated difference (DCD) and find motion vector for aiding prediction among frames. Normally, this method is computed for fixed-size blocks rather than pixels or features because of being simple to implement. However, the fixed-size block matching (FSBM) has numerous drawbacks. For instance, as a result of having used a single disparity for each block, inaccurate disparity estimation and compensation can result. Moreover, the accuracy of the disparity vector may not be sufficient to describe the actual disparity in consequence of integer-pixel resolution. Finally, FSBM tends to produce artifacts at the block boundaries in the predicted image, especially when low rate coding is used.

To overcome this problem, Overlapped Block Motion Compensation (OBMC) has been taken to reduce blocking artifacts and motion compensation error as well with no change in search range and no extra side information by linearly combining the predictions generated using multiple motion vectors, including a block’s motion vector as well as its neighbors. While Overlapped Block Disparity Compensation (OBDC) expanded from OBMC improves smoothness constraints into the disparity estimation.

OBMC is derived as a linear estimator of each pixel intensity. A pixel in each block is predicted based on its motion vector and its neighbor blocks motion vectors. The predicted value, $X_{pre}$ is determined by
\[ X_{\text{pre}}(i, j) = \sum_{\text{all neighboring blocks}} W \cdot X(i - dx, j - dy) \] (2.1)

Where, \( W \) is a weighting coefficient assigned to the pixel. The weighting coefficients should be inversely proportional to the distance between \((i, j)\) and the center position of neighbor block. Figure 2.6 presents an example of the overlapped block motion compensation.

![Overlapping block motion compensation](image)

**Figure 2.6: Overlapping block motion compensation (Thanapirom, 2002)**

### 2.5 Embedded Block Coding with Optimized Truncation (EBCOT)

Video can be considered as the image sequences that can be compressed by using the proper algorithm for each frame. One of the well-known image compression standards is JPEG2000 using the state-of-the-art wavelet technology and the EBCOT algorithm that offers compression performance along with attractive properties such as SNR scalability, resolution scalability and random access capability.

The EBCOT algorithm is related in various degrees to much earlier work on scalable image compression. Some early predecessors are EZW (embedded zero-tree wavelet compression) algorithm (Shapiro, 1993) and SPIHT (spatial partitioning of images into hierarchical trees) algorithm (Said and Pearlman, 1996) that also use wavelet transform algorithm but offer only SNR scalability.

The EBCOT Algorithm, by using DWT, decomposes image into sub-band samples that make compression had resolution scalable property. The result coefficients of transformed image are quantized and partitioned into small code-blocks. The bit-plane coder has coded each code-block independently. After all the sub-band samples have been compressed, the rate-distortion algorithm is applied, so called post-compression rate distortion (PCRD), by truncating each of the independent code-block bit-stream in an optimum way in order to minimize distortion subject to the specific bit-rate. Furthermore, because of only one compression, the advantage of PCD optimization is its reduced complexity. Moreover, there is no need to buffer the entire image or indeed and quantity comparable to the size of the image (Taubman, 2000).

The structures of encoder and decoder are shown in Figure 2.7. The decoder structure essentially mirrors that of the encoder. That is, with the exception of rate control, there is one-to-one correspondence between functional blocks in the encoder and decoder.
2.5.1 Discrete Wavelet Transform (DWT)

Through the application of the wavelet transform, a component is split into numerous frequency bands (i.e., sub-bands). Due to the statistical properties of these sub-band signals, the transformed data can usually be coded more efficiently than the original untransformed data.

The DWT is an important tool in the construction of resolution-scalable bit stream. The DWT is implemented using appropriately designed quadrature mirror filters (QMFs) or biorthogonal filters. The filter consists of a pair of high-pass and low-pass filters. As shown in Figure 2.8, a first DWT stage decomposes the image into four sub-bands, denoted LL₁, HL₁ (horizontally high-pass), LH₁ (vertically high-pass) and HH₁. The next DWT stage decomposes this LL₁ sub-band into four more sub-band, denoted LL₂, HL₂, LH₂ and HH₂. The process continues until the LL₀ band which is the lowest resolution level consisting of the single LL sub-band is obtained. Each successive resolution level, Lₙ, contains the additional sub-bands, which are required to reconstruct the image with twice the horizontal and vertical resolution.

Figure 2.8: Wavelet decomposition structure for three decomposition levels
(Adapted from Taubman, 2000)

On the other words, it is obvious that there are similarities of wavelet coefficients across the wavelet bands. Every coefficient at a given scale, except that of in the lowest frequency band, is related to a set of coefficients of similar orientation at the next finer scale. The coefficient at the coarse scale is called the parent, and all coefficients at the
same spatial location, and of similar orientation at the next finer scale, are called children. By exploiting this idea, the significant compression ratio can be achieved. Figure 2.9 shows the parent-child relationship of wavelet coefficients from a three-decomposition level.

\[ V(x; y) = \left\lfloor \frac{|U(x; y)|}{\Delta_b} \right\rfloor \text{sgn}U(x; y) \quad (2.2) \]

where \( \Delta_b \) is the quantizer step size, \( U(x; y) \) is the input sub-band signal, and \( V(x; y) \) denotes the output quantizer indices for the sub-band. The step size for each sub-band is specified in terms of an exponent, \( \varepsilon_b \), and a mantissa, \( \mu_b \), (Yu Hen Hu, 2001) where

\[ \Delta_b = 2^{-\varepsilon_b} \left( 1 + \frac{\mu_b}{2^{11}} \right) \quad 0 \leq \mu_b < 2^{11} \]

\[ 0 \leq \varepsilon_b < 2^5 \quad (2.3) \]

Figure 2.9: The parent-child relationship of wavelet coefficients (Adapted from SingAREN, 2001)

The scalable compression refers to the generation of a bit-stream which contains embedded subsets, each of which represents an efficient compression of the original image at a reduced resolution or increased distortion.

2.5.2 Quantization/Dequantization

In the encoder, after the tile-component data has been transformed, the resulting coefficients are quantized. Quantization allows greater compression to be achieved, by representing transform coefficients with only the minimal precision required to obtain the desired level of image quality. Quantization of transform coefficients is one of the two primary sources of information loss in the coding path (the other source being the discarding of coding pass data).

Transform coefficients are quantized using scalar quantization with a deadzone. A different quantizer is employed for the coefficients of each sub-band, and each quantizer has only one parameter, its step size. Mathematically, the quantization process is defined as

\[ V(x; y) = \left\lfloor \frac{|U(x; y)|}{\Delta_b} \right\rfloor \text{sgn}U(x; y) \quad (2.2) \]

where \( \Delta_b \) is the quantizer step size, \( U(x; y) \) is the input sub-band signal, and \( V(x; y) \) denotes the output quantizer indices for the sub-band. The step size for each sub-band is specified in terms of an exponent, \( \varepsilon_b \), and a mantissa, \( \mu_b \), (Yu Hen Hu, 2001) where

\[ \Delta_b = 2^{-\varepsilon_b} \left( 1 + \frac{\mu_b}{2^{11}} \right) \quad 0 \leq \mu_b < 2^{11} \]

\[ 0 \leq \varepsilon_b < 2^5 \quad (2.3) \]
In the decoder, the dequantization tries to undo the effects of quantization. Unless all of the quantizer step sizes are less than or equal to one, the quantization process will normally result in some information loss, and this inversion process is only approximate. Mathematically, the dequantization process is defined as

\[
U(x; y) = (V(x; y) + r \text{sgn}(V(x; y))) \Delta_b
\]

where \( \Delta_b \) is the quantizer step size, \( r \) is a bias parameter, \( V(x; y) \) are the input quantizer indices for the sub-band, and \( U(x; y) \) is the reconstructed sub-band signal. Although the value of \( r \) is not normatively specified in the standard, it is likely that many decoders will use the value of one half (Adams, 2002).

### 2.5.3 Tier-1 Encoder

The part of the encoder is separated into Tier-1 encoder and Tier-2 encoder that operate for coding blocks of sub-band samples to be Embedded block bit-streams by Tier-1 encoder and Full-featured bit-stream by Tier-2 encoder as shown in Figure 2.10.

**Figure 2.10: Two-tiered coding structure (Taubman, 2000)**

In the Tier-1 encoder, the quantizer indices are partitioned into code-blocks which are rectangular in shape. Then, each code-block is independently coded by a context-based adaptive arithmetic encoder which can be separated into two steps; Context Formation (CF) and Arithmetic Encoder (AE). The coefficients of a code-block are coded bit-plane by bit-plane from most significant bit-plane (MSB) to less significant bit-plane (LSB). CF scans all bits within a bit-plane of a code-block in a specific order, and generates corresponding contexts for each bit by checking the status of its neighborhood bits. Next, the AE encodes each bit along with the adaptively estimated probabilities from its contexts. The outputs of the arithmetic encoder are the embedded block bit-streams. To improve embedding, fractional bit-plane coding method is used. Embedded coding, which is useful for scalability and for efficient rate control, is actually one of the main features of JPEG 2000. The embedded block bit-streams are comprised of three coding passes per bit plane; Significance Pass, Refinement Pass and Cleanup Pass. The rate and distortion information are also calculated for each pass for subsequent tier-2 processing.

### 2.5.4 Tier-2 Encoder

In the Tier-2 encoder, the coding pass information is packaged into data units called packets comprising of two parts: a header and body, in a process referred to as packetization. The header indicates which coding passes are included in the packet, while the body contains the actual coding pass data itself. One packet is generated for each component, resolution level, layer, and precinct 4-tuple which is a grouping of code-blocks within a sub-band.
Each coding pass is either assigned to one of the $L$ layers or discarded (truncated) that make some information loss under the rate-distortion optimum representation. The coding passes containing the most important data are included in the lower layers, while the coding passes associated with finer details are included in the higher layers. This achieves SNR scalability since the reconstructed image quality improves incrementally with each successive layer processed in the decoder. Figure 2.11 illustrates the layered bit-stream concept. Only five layers are shown with seven code-blocks, for simplicity. The shaded region identifies the block contributions which are discarded by truncating the bit-stream between layer 3 and 4.

![Figure 2.11: Illustration of block aid to bit-stream layers (Taubman, 2000)](image)

Each quality layer must include auxiliary information to identify the size of each code-block’s contribution to the layer. When the number of layers is large, only a subset of the code-blocks will contribute to any given layer, introducing substantial redundancy in this auxiliary information.

Finally, the output of the Tier-2 encoder is the full-featured bit-stream generated by concatenating the suitably truncated representations of each code-block including auxiliary information to identify the truncate point and the corresponding length. For more details about the Tier-1 and Tier-2 encoders, the reader is referred to Appendix A.

2.5.5 Rate Control

A rate control utilizes for managing the specific used bits of the output-compressed stream. The EBCOT coding uses the rate distortion optimization for controlling the bit used in each coding block. For rate distortion optimization, the encoder knows the contribution that each compressed pass makes to bit-rate, and can also calculate the distortion reduction associated with each compressed pass. Using this information, the encoder can include the compressed passes in order of decreasing distortion reduction per unit rate until the bit budget has been exhausted. This approach is very flexible in that different distortion metrics can be easily accommodated (e.g., mean squared error, visually weighted mean squared error, etc.). Moreover, it can achieve a desired bit-rate in a single iteration with minimum distortion. For more detail treatment of rate control, the reader is referred to Appendix A.
2.6 Picture Quality Assessment

Conversion of digital pictures from one format to another format, as well as their compression for bit rate reduction, introduces some distortions. It is necessary to know whether the introduced distortion is acceptable to the viewers. Traditionally, the two most common schemes of picture quality measurement are mean squared error (MSE) and peak signal to noise ratio (PSNR). The MSE is simply the mean of the squared error at each pixel location. Let $X = (x_1, x_2, \ldots, x_n)$ represent the values of the $n$ pixels of an image, and $Y = (y_1, y_2, \ldots, y_n)$ represent the reconstructed image. Then the MSE is defined as (Ghanbari, 1999)

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (x_i - y_i)^2$$  \hspace{1cm} (2.5)

The higher values of MSE correspond to lower image quality and the lower values of MSE give better picture quality. The PSNR is defined as (Ghanbari, 1999)

$$\text{PSNR} = 10 \log \left[ \frac{M^2}{\text{MSE}} \right]$$  \hspace{1cm} (2.6)

where $M$ denotes the maximum possible value for any pixel. For an example 8 bit monochrome images, $M = 255$. The PSNR is measured in units of decibels (dB). Note that higher values for the PSNR usually correspond to better image quality and vice versa.
This chapter explains the proposed stereo video coding scheme, including the significant parameters.

3.1 Proposed System Model

The proposed stereo video encoder is illustrated in Figure 3.1. It composes of two sub-encoders; Main Stream Encoder for coding left frames of video sequences and Auxiliary Stream Encoder for coding right frames of the video sequences. A rate control utilizes for managing the specific used bits of output-compressed stream.

![Image of proposed stereo video encoder](image)

Figure 3.1: The proposed stereo video encoder (Haskell et al., 1996)

The compression scheme runs in a group of pictures (GOP), I-frame, P-frames and B-frames as in MPEG-2 video coding standard. The association of each frame is shown in Figure 3.2. Note that I, P and B pictures of the main stream are represented by $I_M$, $P_M$ and $B_M$ whilst I, P and B pictures of the auxiliary stream are represented by $I_A$, $P_A$ and $B_A$. The 12- GOP size and 3- sub GOP size are selected for the temporal picture structures of both main stream and auxiliary stream. It is noticeable that theGOP structure in this thesis is not similar the one appearing in MPEG-2 (shown in Figure 2.4) as the B-pictures in the auxiliary stream can be predicted tri-directional while those of in MPEG-2 are predicted bi-directional. This results in high prediction efficiency. The next parts describe the main stream encoder and the auxiliary stream encoder.

3.1.1 Main Stream Encoder

The main stream encoder is presented in Figure 3.3. The left streams of stereo video sequences are encoded as intra or inter frames. The intra frame is applied to EBCOT encoder operating as shown in Figure 2.7. In each frame, pixels are transformed into Discrete Wavelet Transform (DWT) coefficients, quantized and coded by the Tier-1 and Tier-2 encoders. The EBCOT decoder is applied to convert frames back for predicting frames. Then, the predicting frame is applied to the block motion estimation as well as sent to the auxiliary stream encoder. The motion vectors, the outputs of a block motion estimation, both horizontal and vertical vectors are encoded by Huffman coding. The overlapped block motion compensation (OBMC) is used to predict image from block motion compensation in order to reduce blocking artifacts.
Figure 3.2: Prediction process in stereo encoder (Thanapirom, 2002)

Figure 3.3: Main Stream Encoder (Adapted from Thanapirom, 2002)

Figure 3.4: Auxiliary Stream Encoder (Adapted from Thanapirom, 2002)
3.1.2 Auxiliary Stream Encoder

The auxiliary stream encoder in Figure 3.4 is mostly similar to the main stream encoder. It composes of the disparity estimation and compensation block for predicting the auxiliary frame (or DCP) from the decoded left stream (main stream) and the motion estimation and compensation block for motion compensated prediction (MCP) among right frames (auxiliary frames) as shown in Figure 3.2. Before applying the EBCOT encoder to an auxiliary image, the best estimation is chosen among two Displaced Frame Difference (DFD) images, which are the output of the disparity/motion estimation and compensation blocks, and the direct right frame. The EBCOT decoder is applied to convert frames back for predicting frames and enter to the motion estimation block. The selected image either DFD or direct right image is coded via EBCOT scheme and the motion or disparity vectors of the best estimation are coded by Huffman coding similar to the main stream encoder and are transmitted together with the coded frames. The rate control does not separately control the bit rate between main and auxiliary streams but it controls the bit rate in proportional to the available bit budget for both streams.

The details of the disparity/motion estimation and compensation blocks in the auxiliary stream encoder are shown in Figure 3.5. The regenerated frame, the decoded left frame or the decoded reference right frame, is used for computing disparity or motion vectors and the OBDC/OBMC is performed so as to get the predicted image. The residual can be constructed by subtracting the anchor image from the predicted image.

![Diagram](image.png)

**Figure 3.5: Disparity/motion estimation and compensation in the auxiliary stream encoder (Thanapirom, 2002)**

The DFD images (or prediction error images or residual frames), the outputs of disparity/motion estimation and compensation blocks, and the direct right frame compete each other at the choosing block. The reason of the supplement of the direct right frame can be described as follows.

Ordinarily, the higher the energy or information in which the picture contains, the more the bits used for image coding. Though the disparity/motion estimation and compensations generally tend to give the lower-energy DFD images, these images may carry more energy and use the entire coded bits more than those of the original right images. Figure 3.6, for an example, show the right image and the prediction error image of the successive frames of “Aqua” standard test stereo image. It can be noticed that the prediction error image still carries much information.
Figure 3.6: Aqua stereo image pair (DESITMA Stereo Image) (a) Right Image (b) Prediction Error Image

On that account, the original right frames are added to be one alternative for better compression. The criterion of selection is weighted on the total energy of picture carried or the variation of information of pictures following the equation 3.1.

\[ Var(X) = \frac{1}{n} \sum_{i=1}^{n} (x_i - \bar{X})^2 \]  

where \( X = (x_1, x_2, \ldots, x_n) \) represents the values of the \( n \) pixels of an image, and \( \bar{X} \) represents the mean of \( X \). The frame with lesser value, either the DFD images from the disparity/motion estimations or the direct right image, will be selected.

In most cases, the both stereo streams are not necessary to be coded with the same quality since the quality of one of the views can be significantly lower when compared to the other view without sustaining perceivable visual distortions. This fact is supported by many practical experiments (Haskell et al., 1996). The experiments claim that the quality of one view can be lower by 3 dB or more as compared to the other view as long as it can preserve disparity edges. Further support for unequal distribution of quality between the two views is provided by the leading eye theory. It presents an idea that the overall subjective impression in stereoscopic 3D viewing depends mainly on the quality perceived.
by the leading eye of viewers. Such that for a given bit rate, significant benefits can be
grown from unequal distribution of bits between the two views.

3.2 System Specifications

3.2.1 Motion Estimation

The motion estimation is the block-based algorithm that divides the anchor frame
into nonoverlapping 16 by 16 blocks following the MPEG-2 standard. This choice is a
result of the trade-off between the coding gain provided by using motion information and
the overhead needed to represent it. For each block in the anchor frame and a search range
of ±16 pixels in both directions, the predicted block is found by minimizing the sum of
absolute difference (SAD). To favour zero motion vectors, the SAD of zero displacement
is reduced by a value of 100 (Martucci et al., 1997). The half-pel refinement is also
exploited in all four directions using bilinear interpolation of the eight surrounding pixels
in order to diminish the mismatch error. The motion estimation is performed only in the
luminance component. For the chrominance components, motion vectors are divided by
two and quarter-pel interpolation is done to obtain the prediction of the chromic blocks.
The motion vector is independently coded by Huffman Coding which reduces the average
code length used to represent the symbols of an alphabet. Symbols of the source alphabet
which occur frequently are assigned with short length codes.

3.2.2 Overlapped Block Motion Compensation

A significant drawback of DWT based coding when compared to DCT coding is
that one has to sacrifice many bits to code the artificial high frequency information at the
block boundaries of motion compensated P- or B- frames. Therefore OBMC is employed
to offer substantial reductions in the prediction error. OBMC predicts the current frame by
repositioning overlapping blocks of pixels from the previous frame, each weighted by a
smoothing window. In this method, each 8 by 8 block in a macroblock is overlapped with
three major adjacent blocks that do not belong to the same macroblock. In the proposed
algorithm, a Raised Cosine window which shows the best performance compared to
Bilinear window will be used. The Raised Cosine window \( w \) (Orchard and Sullivan,
1994) is given by

\[
w(n, m) = w_n \cdot w_m, \quad w_m = \frac{1}{2} \left[ 1 - \cos \left( \frac{n + \frac{1}{2}}{16} \right) \right]
\]

for \( n = 0, \ldots, 31 \) (3.2)

3.2.3 Disparity Estimation and Compensation

The disparity estimation and compensation will be used mostly similar to the
motion estimation and compensation scheme of the motion estimation and compensation.
However, by using chromatic components of the images to capture the objects and indicate
the displacement of the successive stereo pair, the variable scanning range of disparity
vector calculation can be applied. Besides, it can assume that the sequences are received
from parallel camera configuration thus the disparity has only a horizontal component. The
search window size can be used in three sizes; 0 to 8, 0 to 16 and 0 to 32 in horizontal
directions for variable scanning range and 0 to 16 for uniform scanning range as well as ± 4 in vertical direction, when using the left frame as the reference frame. The reason for having a search range in vertical direction comes from the fact that the camera axes are not always parallel, and the occlusion junctions in a stereo image pair may be different in vertical position as well. For a more detailed treatment about the variable scanning range of disparity estimation, the reader is referred to Appendix B.

3.2.4 EBCOT Encoder

In this proposed scheme, all parameters used in the encoder based on EBCOT algorithm are followed the JPEG2000 standard (M. D. Adams, 2002). The irreversible transform or real mode operation is selected for wavelet transform because of the capability of lossy compression. Hence, the Daubechies 9/7 filter is employed. The analysis and corresponding synthesis filter coefficients of Daubechies 9/7 is given in Table 3.1. DWT is implemented up to 4 resolution levels in luminance component and up to 3 resolution levels in chrominance components.

Table 3.1: Daubechies 9/7 analysis and synthesis filter coefficients. (Rabbani and Joshi, 2002)

<table>
<thead>
<tr>
<th>i</th>
<th>Lowpass Filter $h_L(i)$</th>
<th>Highpass Filter $h_H(i)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.6029490182363579</td>
<td>1.115087052456994</td>
</tr>
<tr>
<td>±1</td>
<td>0.2668641184428723</td>
<td>-0.5912717631142470</td>
</tr>
<tr>
<td>±2</td>
<td>-0.07822326652898785</td>
<td>-0.05754352622849957</td>
</tr>
<tr>
<td>±3</td>
<td>-0.01686411844287495</td>
<td>0.09127176311424948</td>
</tr>
<tr>
<td>±4</td>
<td>0.02674875741080976</td>
<td></td>
</tr>
</tbody>
</table>

For quantization, in the case of real mode, the quantizer step sizes are chosen in conjunction with the rate control. The quantization step size $\Delta$ is represented relative to the dynamic range of sub-band that corresponds to Figure 3.7.

![Figure 3.7: The uniform scalar quantization with dead zone (Skodras et. al., 2000)](image-url)
As one might expect, the quantizer step sizes used by the encoder are conveyed to the decoder via the code stream. In passing, the step sizes specified in the code stream are relative and not absolute quantities. That is, the quantizer step size for each band is specified relative to the nominal dynamic range of the sub-band signal.

### 3.2.5 Rate Control

The number of bits to be spent for each frame is determined based on the specific bit allocation for that type of frame and the number of bits available.

As mentioned above, real mode, which is proposed first, uses two methods to control the bit rate. The first one is deciding on proper quantization step size and, secondly, selecting the subset of coding pass to include in the code stream. To discard coding passes in order to control the rate, the encoder knows the contribution that each coding pass makes to the rate, and can also calculate the distortion reduction associated with each coding pass. Using this information, the encoder can then include the coding passes in order of decreasing distortion reduction per unit rate until the bit budget has been achieved.

The target bits for each frame in stereo stream are calculated based on this complexity and after coding each frame the complexity are updated according to the new bit rate. The PSNR difference between the main stream and the auxiliary to about 3 dB can be forced according to the leading eye theory. The simulation operates at various bit rate in order to reflect on the improving or degrading tendency of the proposed stereo video codec by which the intra-frame coding and residual frame coding encoded via EBCOT against DCT based coded and ZTE based codec.

### 3.3 Enhancement Techniques for Stereoscopic Compression

This thesis has introduced two enhancement techniques for further improve the efficient compression for stereoscopic system. These two algorithms spend the benefit information of a stereo pair so as to preclude the imperfection of the conventional schemes.

First, the variable scanning range for disparity estimation has been proposed in order to spread out the scanning window that may be limited by the traditional scheme, the uniform scanning range. Generally, the uniform window is chosen for convenient operation and its size is selected by trade-off between the coding gain provided by using disparity information and overhead needed to represent it. However, the foregrounds in the stereo images do not have much disparity, while some objects in far distance require more scanning span for searching the most matching block to diminish error values of residual frame. Therefore, the extensive scanning in horizontal direction permits the upgrading of disparity prediction and it is proposed in the uncomplicated scheme as referred in Appendix B.

Second, the occlusions in a stereo pair have been considered, since the occluded areas may cause the erroneous matching in disparity estimation and spend many waste bits for encoding. The occluded prediction and independently coding via EBCOT algorithm of occluded regions are given an explanation in Appendix C.
CHAPTER 4
RESULTS AND DISCUSSIONS

This chapter illustrates the simulation results done throughout this thesis study. First of all, the proposed codec was tested with the gray scale stereo image pairs and investigated the performance of EBCOT coding compared with the existing stereoscopic standard codec, MPEG-2, which is based on DCT and one high effective coding, ZTE, applied for stereo images. The objective image quality is measured in the term of peak-to-peak signal to noise ratio (PSNR) and the compression efficiency is measured in bit per pixel (bpp). All of test images were taken from DISTIMA Stereoscopic Image Pairs.

To receive further compression efficiency, the colour features of images have been considered to segment the region-of-interest and to predict the occlusion of the stereo image pairs. The performance of the proposed scheme is also investigated and compared with the ordinary scheme. The image coding results could be interpreted as the performance of coding I-frames in the main stream and the auxiliary stream.

Subsequently, all proposed schemes were performed with the stereo video sequences and evaluated the performance against that of DCT based codec and ZTE based codec. All of test sequences were taken from AVDS Sequences.

4.1 Results for the Gray Scale Stereo Image Pairs

The simulation results are taken over three gray scale standard stereo image pairs, namely Aqua, Tunnel and Manege to examine the performance of EBCOT coding compared with that of DCT coding and ZTE coding for stereo images. The left images were coded independently and the right images were coded by using uniform scanning range for disparity estimation and compensation which is the same process using in the auxiliary stream encoder. All parameters for this simulation are same as used in the proposed stereo encoder for Y-component. The simulation was tried to fix the quality of the reconstructed left images equally in all coding algorithm so as to the rate-distortion performance of the reconstructed right images of each scheme could be clearly compared.

Figure 4.1 reveals the PSNR curves of the reconstructed stereo image pairs in a range of bit rates of three stereo image pairs; Aqual, Tunnel and Manege respectively.
The simulation results show that the EBCOT coding produces PSNR gains of around 1-2 dB as compared to ZTE coding and around 2-3 dB as compared to DCT coding. Figure 4.2 show the output of stereo image coding at 0.25 bpp; the original right image and the decoded right images. By this point, the EBCOT coding show the interesting performance outstandingly and it could lay emphasis on the likelihood of using EBCOT coding to enrich the stereoscopic image/video compression.
Figure 4.2: The right images of ‘Tunnel’ coded at 0.25 bpp for image pair. The original right image illustrated in (a) and its reconstructed images from (b) DCT coding (c) ZTE coding (D) EBCOT coding
4.2 Results for the Colour Stereo Image Pairs

Typically, binocular disparity estimation and compensation is used for the removal of the inter-frame redundancies by predicting in the luminance domain. Nevertheless, the chromatic information can be exploited for disparity estimation, which results in an increased efficiency of stereo image compression as shown in Figure 4.3 and Figure 4.4.

The proposed algorithm lies in the fact that it uses chrominance information of a stereo image pair to help accurately identify stereoscopically distinctive objects in a 3D scene. It subsequently uses these separate objects to more efficiently and selectively code regions of particular interest and identify the maximum variable scanning range for disparity estimation while maintaining the overall image quality higher compared to using conventional techniques.
Figure 4.3: The rate-distortion performance of reconstructed right images illustrated in RGB colour domain. (a) Aqua (b) Tunnel (c) Manege

In this simulation, the performance of the proposed scheme, variable scanning range for disparity estimation by using colour information, was compared with the traditional technique, uniform scanning range for disparity estimation. The bit rate was levelly controlled to preserve PSNR of the decoded left images 3 dB more than that of the decoded right images of uniform scanning range method, whilst the quality of reference images (left images) was maintained equally for both algorithms. The rate-distortion curves in Figure 4.3 illustrates the improvement around 0.5-1 dB in all components; Red, Green and Blue component, of the proposed scheme against the traditional scheme which both were coded via EBCOT coding. Figure 4.4 displays the examples of the reconstructed right images for subjective quality assessment.
Figure 4.4: The right images of ‘Manege’ coded at 0.25 bpp for image pair. The original right image illustrated in (a) and its reconstructed images from (b) Uniform scanning range (c) Variable scanning range

4.3 Results for Concerning the Occlusion in Stereo Image Pair

As using the decoded image of one side of the stereo image pair to be the reference image, the occlusions appearing in another side image are the regions that cannot be seen in the reference image. Hence, these might cause the erroneous matching in the prediction algorithm, especially in block based method. If the occlusions can be predicted accurately and coded independently from the conventional scheme, the performance of the novel codec might be enhanced.

In this thesis, the proposed algorithm for occluded prediction, which is clearly explained in Appendix C, was carried out with two views of multi-view test images; Claude2 and Claude4. Afterward, the scheme was applied to the first frame of Booksale and Crowd sequences.
The simulation results obviously show the slight improvement of the proposed scheme over the conventional scheme only if the high bit rate is employed. At low bit rate, the overhead identified the occlusion might not be compensated by the better quality of reconstructed images.

4.4 Results for Stereo Video Sequences

The previous sections discussed in stages the simulation results of the proposed scheme for stereo image pair. Consequently, the proposed scheme was applied to stereo video sequences and the performance of decoded stereo sequences was measured to verify the effectiveness of the proposed stereoscopic video codec against that of DCT coding and ZTE coding.

The variable scanning range for disparity estimation was exploited only in I-frames of each GOP, since the surplus overhead would significantly affect in very low bit-rate coding, especially in B-frames, as the results shown in Figure 4.3, the performance gap is more taper in lower rate. Likewise, the scheme for predicting the occlusion region was applied only in I-frames, because P-frames and B-frames in auxiliary stream are predicted multidirectional. In other words, the occluded region in P-frame and B-frame of the auxiliary stream could be matched with some parts of the reference frames in the same
stream. Besides, from the simulation results in section 4.3, the overhead for occluded identification cannot be compensated by the improvement of coding performance in low bit rate; therefore, the proposed scheme for coding occlusion in stereo video sequences was exploited only in high bit rate. The high bit rate is defined if the bit rate is equal or more than 1.00 bpp. Note that these enhanced methods were applied in all coding schemes for fairing comparison.

In trial simulations, two stereo video sequences were coded, namely *Booksale* and *Crowd*, totally 85 frames in YUV colour format. The numerical results obtained four bit rates, 0.244 bpp, 0.34 bpp, 0.472 bpp and 1.00 bpp, were tabulated in Table 4.1 - Table 4.6.

Table 4.1: The average PSNRs of *Booksale* sequences for the different type of pictures at 0.244 bpp

<table>
<thead>
<tr>
<th>Frame type</th>
<th>Component</th>
<th>Coding</th>
<th>Y (dB)</th>
<th>U (dB)</th>
<th>V (dB)</th>
<th>Y (dB)</th>
<th>U (dB)</th>
<th>V (dB)</th>
<th>avg. bits (kbits)</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>DCT</td>
<td></td>
<td>33.85</td>
<td>42.41</td>
<td>42.47</td>
<td>30.71</td>
<td>40.18</td>
<td>40.54</td>
<td>153.32</td>
</tr>
<tr>
<td></td>
<td>ZTE</td>
<td></td>
<td>34.84</td>
<td>44.27</td>
<td>44.34</td>
<td>31.86</td>
<td>41.96</td>
<td>42.30</td>
<td>151.43</td>
</tr>
<tr>
<td></td>
<td>EBCOT</td>
<td></td>
<td>35.80</td>
<td>45.02</td>
<td>45.59</td>
<td>32.86</td>
<td>41.27</td>
<td>41.88</td>
<td>149.72</td>
</tr>
<tr>
<td>P</td>
<td>DCT</td>
<td></td>
<td>32.70</td>
<td>40.58</td>
<td>40.79</td>
<td>29.43</td>
<td>36.21</td>
<td>36.65</td>
<td>73.93</td>
</tr>
<tr>
<td></td>
<td>ZTE</td>
<td></td>
<td>33.19</td>
<td>40.48</td>
<td>40.64</td>
<td>30.96</td>
<td>36.95</td>
<td>37.33</td>
<td>67.49</td>
</tr>
<tr>
<td></td>
<td>EBCOT</td>
<td></td>
<td>34.62</td>
<td>40.76</td>
<td>41.08</td>
<td>31.29</td>
<td>36.97</td>
<td>37.05</td>
<td>69.75</td>
</tr>
<tr>
<td>B</td>
<td>DCT</td>
<td></td>
<td>33.02</td>
<td>39.39</td>
<td>39.58</td>
<td>29.66</td>
<td>37.23</td>
<td>37.75</td>
<td>49.24</td>
</tr>
<tr>
<td></td>
<td>ZTE</td>
<td></td>
<td>34.05</td>
<td>39.09</td>
<td>39.33</td>
<td>30.97</td>
<td>37.20</td>
<td>37.77</td>
<td>44.75</td>
</tr>
<tr>
<td></td>
<td>EBCOT</td>
<td></td>
<td>34.35</td>
<td>39.12</td>
<td>39.55</td>
<td>31.05</td>
<td>37.01</td>
<td>37.07</td>
<td>45.22</td>
</tr>
</tbody>
</table>

Table 4.2: The average PSNRs of *Crowd* sequences for the different type of pictures at 0.244 bpp

<table>
<thead>
<tr>
<th>Frame type</th>
<th>Component</th>
<th>Coding</th>
<th>Y (dB)</th>
<th>U (dB)</th>
<th>V (dB)</th>
<th>Y (dB)</th>
<th>U (dB)</th>
<th>V (dB)</th>
<th>avg. bits (kbits)</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>DCT</td>
<td></td>
<td>33.68</td>
<td>42.66</td>
<td>43.5</td>
<td>31.22</td>
<td>40.20</td>
<td>41.05</td>
<td>173.21</td>
</tr>
<tr>
<td></td>
<td>ZTE</td>
<td></td>
<td>34.77</td>
<td>44.52</td>
<td>45.43</td>
<td>31.32</td>
<td>41.78</td>
<td>42.58</td>
<td>162.05</td>
</tr>
<tr>
<td></td>
<td>EBCOT</td>
<td></td>
<td>35.39</td>
<td>44.96</td>
<td>46.31</td>
<td>32.48</td>
<td>41.65</td>
<td>42.19</td>
<td>157.26</td>
</tr>
<tr>
<td>P</td>
<td>DCT</td>
<td></td>
<td>32.70</td>
<td>40.80</td>
<td>41.48</td>
<td>29.91</td>
<td>38.19</td>
<td>38.82</td>
<td>82.66</td>
</tr>
<tr>
<td></td>
<td>ZTE</td>
<td></td>
<td>33.83</td>
<td>40.08</td>
<td>41.05</td>
<td>30.82</td>
<td>37.83</td>
<td>38.63</td>
<td>78.36</td>
</tr>
<tr>
<td></td>
<td>EBCOT</td>
<td></td>
<td>34.13</td>
<td>40.86</td>
<td>41.8</td>
<td>30.68</td>
<td>37.81</td>
<td>38.45</td>
<td>77.51</td>
</tr>
<tr>
<td>B</td>
<td>DCT</td>
<td></td>
<td>32.27</td>
<td>39.25</td>
<td>40.02</td>
<td>30.16</td>
<td>37.60</td>
<td>38.33</td>
<td>47.40</td>
</tr>
<tr>
<td></td>
<td>ZTE</td>
<td></td>
<td>33.79</td>
<td>39.96</td>
<td>40.86</td>
<td>30.45</td>
<td>37.20</td>
<td>38.15</td>
<td>46.51</td>
</tr>
<tr>
<td></td>
<td>EBCOT</td>
<td></td>
<td>34.05</td>
<td>38.60</td>
<td>39.45</td>
<td>31.04</td>
<td>37.05</td>
<td>37.56</td>
<td>49.36</td>
</tr>
</tbody>
</table>
Table 4.3: The average PSNRs of *Booksale* sequences for the different type of pictures at 0.34 bpp

<table>
<thead>
<tr>
<th>Frame type</th>
<th>Coding</th>
<th>Component</th>
<th>Main stream (dB)</th>
<th>Auxiliary stream (dB)</th>
<th>avg. bits (kbits)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DCT</td>
<td>Y</td>
<td>36.95</td>
<td>44.70</td>
<td>44.94</td>
</tr>
<tr>
<td></td>
<td></td>
<td>U</td>
<td>33.35</td>
<td>41.50</td>
<td>41.84</td>
</tr>
<tr>
<td></td>
<td></td>
<td>V</td>
<td>33.95</td>
<td>43.43</td>
<td>43.93</td>
</tr>
<tr>
<td></td>
<td>ZTE</td>
<td>Y</td>
<td>38.11</td>
<td>47.08</td>
<td>47.38</td>
</tr>
<tr>
<td></td>
<td></td>
<td>U</td>
<td>35.12</td>
<td>43.66</td>
<td>43.88</td>
</tr>
<tr>
<td></td>
<td>EBCOT</td>
<td>Y</td>
<td>38.11</td>
<td>47.08</td>
<td>47.38</td>
</tr>
<tr>
<td></td>
<td></td>
<td>U</td>
<td>35.12</td>
<td>43.66</td>
<td>43.88</td>
</tr>
</tbody>
</table>

Table 4.4: The average PSNRs of *Crowd* sequences for the different type of pictures at 0.34 bpp

<table>
<thead>
<tr>
<th>Frame type</th>
<th>Coding</th>
<th>Component</th>
<th>Main stream (dB)</th>
<th>Auxiliary stream (dB)</th>
<th>avg. bits (kbits)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DCT</td>
<td>Y</td>
<td>36.44</td>
<td>44.47</td>
<td>45.26</td>
</tr>
<tr>
<td></td>
<td></td>
<td>U</td>
<td>32.70</td>
<td>40.98</td>
<td>41.78</td>
</tr>
<tr>
<td></td>
<td></td>
<td>V</td>
<td>33.75</td>
<td>42.90</td>
<td>43.72</td>
</tr>
<tr>
<td></td>
<td>ZTE</td>
<td>Y</td>
<td>37.56</td>
<td>46.36</td>
<td>47.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>U</td>
<td>33.00</td>
<td>42.90</td>
<td>43.72</td>
</tr>
<tr>
<td></td>
<td>EBCOT</td>
<td>Y</td>
<td>37.49</td>
<td>46.78</td>
<td>47.81</td>
</tr>
<tr>
<td></td>
<td></td>
<td>U</td>
<td>33.75</td>
<td>42.90</td>
<td>43.72</td>
</tr>
<tr>
<td></td>
<td></td>
<td>V</td>
<td>34.75</td>
<td>43.37</td>
<td>44.12</td>
</tr>
<tr>
<td></td>
<td>DCT</td>
<td>Y</td>
<td>34.89</td>
<td>41.94</td>
<td>42.78</td>
</tr>
<tr>
<td></td>
<td></td>
<td>U</td>
<td>31.08</td>
<td>39.64</td>
<td>40.34</td>
</tr>
<tr>
<td></td>
<td></td>
<td>V</td>
<td>32.45</td>
<td>39.97</td>
<td>40.77</td>
</tr>
<tr>
<td></td>
<td>ZTE</td>
<td>Y</td>
<td>35.14</td>
<td>42.41</td>
<td>43.26</td>
</tr>
<tr>
<td></td>
<td></td>
<td>U</td>
<td>32.63</td>
<td>39.35</td>
<td>40.17</td>
</tr>
<tr>
<td></td>
<td>EBCOT</td>
<td>Y</td>
<td>35.51</td>
<td>42.07</td>
<td>43.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>U</td>
<td>32.63</td>
<td>39.35</td>
<td>40.17</td>
</tr>
<tr>
<td></td>
<td></td>
<td>V</td>
<td>32.63</td>
<td>39.35</td>
<td>40.17</td>
</tr>
<tr>
<td></td>
<td>DCT</td>
<td>Y</td>
<td>34.9</td>
<td>40.21</td>
<td>41.05</td>
</tr>
<tr>
<td></td>
<td></td>
<td>U</td>
<td>31.77</td>
<td>38.12</td>
<td>38.89</td>
</tr>
<tr>
<td></td>
<td></td>
<td>V</td>
<td>32.28</td>
<td>37.98</td>
<td>38.93</td>
</tr>
<tr>
<td></td>
<td>ZTE</td>
<td>Y</td>
<td>35.29</td>
<td>40.10</td>
<td>40.98</td>
</tr>
<tr>
<td></td>
<td></td>
<td>U</td>
<td>32.28</td>
<td>37.98</td>
<td>38.93</td>
</tr>
<tr>
<td></td>
<td>EBCOT</td>
<td>Y</td>
<td>35.47</td>
<td>39.85</td>
<td>40.89</td>
</tr>
<tr>
<td></td>
<td></td>
<td>U</td>
<td>32.59</td>
<td>37.34</td>
<td>38.29</td>
</tr>
<tr>
<td></td>
<td></td>
<td>V</td>
<td>32.59</td>
<td>37.34</td>
<td>38.29</td>
</tr>
</tbody>
</table>
Table 4.5: The average PSNRs of *Booksale* sequences for the different type of pictures at 0.472 bpp

<table>
<thead>
<tr>
<th>Frame type</th>
<th>Component Coding</th>
<th>Stream</th>
<th>Y (dB)</th>
<th>U (dB)</th>
<th>V (dB)</th>
<th>Y (dB)</th>
<th>U (dB)</th>
<th>V (dB)</th>
<th>avg. bits (kbits)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Main stream</td>
<td></td>
<td></td>
<td></td>
<td>Auxiliary stream</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>DCT</td>
<td>38.99</td>
<td>45.81</td>
<td>46.01</td>
<td>35.54</td>
<td>42.55</td>
<td>42.94</td>
<td>320.13</td>
</tr>
<tr>
<td>I</td>
<td>ZTE</td>
<td>40.3</td>
<td>47.96</td>
<td>48.28</td>
<td>37.38</td>
<td>44.92</td>
<td>45.39</td>
<td>318.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>EBCOT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ZTE</td>
<td>39.92</td>
<td>48.04</td>
<td>48.35</td>
<td>36.48</td>
<td>45.17</td>
<td>45.6</td>
<td>319.08</td>
</tr>
<tr>
<td></td>
<td>EBCOT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P</td>
<td></td>
<td>DCT</td>
<td>36.89</td>
<td>42.91</td>
<td>43.20</td>
<td>33.65</td>
<td>40.04</td>
<td>40.46</td>
<td>159.43</td>
</tr>
<tr>
<td></td>
<td>ZTE</td>
<td>37.35</td>
<td>44.07</td>
<td>44.34</td>
<td>34.87</td>
<td>40.75</td>
<td>41.22</td>
<td>156.03</td>
<td></td>
</tr>
<tr>
<td></td>
<td>EBCOT</td>
<td>38.27</td>
<td>43.67</td>
<td>44.04</td>
<td>35.00</td>
<td>40.67</td>
<td>40.99</td>
<td>164.17</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td></td>
<td>DCT</td>
<td>35.66</td>
<td>42.42</td>
<td>42.79</td>
<td>33.65</td>
<td>39.14</td>
<td>39.64</td>
<td>103.79</td>
</tr>
<tr>
<td></td>
<td>ZTE</td>
<td>36.63</td>
<td>42.53</td>
<td>42.92</td>
<td>34.41</td>
<td>38.59</td>
<td>39.14</td>
<td>99.18</td>
<td></td>
</tr>
<tr>
<td></td>
<td>EBCOT</td>
<td>37.68</td>
<td>42.27</td>
<td>42.80</td>
<td>34.65</td>
<td>38.67</td>
<td>39.04</td>
<td>98.75</td>
<td></td>
</tr>
</tbody>
</table>

Table 4.6: The average PSNRs of *Crowd* sequences for the different type of pictures at 0.472 bpp

<table>
<thead>
<tr>
<th>Frame type</th>
<th>Component Coding</th>
<th>Stream</th>
<th>Y (dB)</th>
<th>U (dB)</th>
<th>V (dB)</th>
<th>Y (dB)</th>
<th>U (dB)</th>
<th>V (dB)</th>
<th>avg. bits (kbits)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Main stream</td>
<td></td>
<td></td>
<td></td>
<td>Auxiliary stream</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>DCT</td>
<td>38.98</td>
<td>45.53</td>
<td>46.25</td>
<td>34.74</td>
<td>39.98</td>
<td>40.80</td>
<td>313.68</td>
</tr>
<tr>
<td>I</td>
<td>ZTE</td>
<td>39.61</td>
<td>44.62</td>
<td>45.61</td>
<td>35.22</td>
<td>40.46</td>
<td>41.27</td>
<td>316.10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>EBCOT</td>
<td>39.77</td>
<td>47.74</td>
<td>49.17</td>
<td>36.70</td>
<td>44.40</td>
<td>45.65</td>
<td>314.25</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ZTE</td>
<td>36.38</td>
<td>42.66</td>
<td>43.49</td>
<td>33.57</td>
<td>38.25</td>
<td>39.12</td>
<td>169.12</td>
</tr>
<tr>
<td></td>
<td>EBCOT</td>
<td>37.78</td>
<td>43.3</td>
<td>44.31</td>
<td>34.88</td>
<td>40.63</td>
<td>41.82</td>
<td>172.23</td>
<td></td>
</tr>
<tr>
<td>P</td>
<td></td>
<td>DCT</td>
<td>36.35</td>
<td>42.28</td>
<td>43.16</td>
<td>34.78</td>
<td>40.05</td>
<td>40.88</td>
<td>171.43</td>
</tr>
<tr>
<td></td>
<td>ZTE</td>
<td>37.78</td>
<td>43.3</td>
<td>44.31</td>
<td>34.88</td>
<td>40.63</td>
<td>41.82</td>
<td>172.23</td>
<td></td>
</tr>
<tr>
<td></td>
<td>EBCOT</td>
<td>37.77</td>
<td>47.74</td>
<td>49.17</td>
<td>36.70</td>
<td>44.40</td>
<td>45.65</td>
<td>314.25</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td></td>
<td>DCT</td>
<td>36.01</td>
<td>42.83</td>
<td>43.70</td>
<td>32.13</td>
<td>38.77</td>
<td>39.66</td>
<td>100.33</td>
</tr>
<tr>
<td></td>
<td>ZTE</td>
<td>36.31</td>
<td>41.88</td>
<td>42.77</td>
<td>33.74</td>
<td>38.96</td>
<td>39.91</td>
<td>100.21</td>
<td></td>
</tr>
<tr>
<td></td>
<td>EBCOT</td>
<td>36.75</td>
<td>41.05</td>
<td>42.15</td>
<td>33.85</td>
<td>38.27</td>
<td>39.05</td>
<td>100.23</td>
<td></td>
</tr>
</tbody>
</table>
Table 4.7: The average PSNRs of *Booksale* sequences for the different type of pictures at 1.00 bpp

<table>
<thead>
<tr>
<th>Frame type</th>
<th>Component Coding</th>
<th>Stream</th>
<th>Main stream (dB)</th>
<th>Auxiliary stream (dB)</th>
<th>avg. bits (kbits)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Y</td>
<td>U</td>
<td>V</td>
</tr>
<tr>
<td>I</td>
<td>DCT</td>
<td></td>
<td>42.24</td>
<td>48.50</td>
<td>48.63</td>
</tr>
<tr>
<td></td>
<td>ZTE</td>
<td></td>
<td>43.69</td>
<td>50.12</td>
<td>51.09</td>
</tr>
<tr>
<td></td>
<td>EBCOT</td>
<td></td>
<td>44.06</td>
<td>52.62</td>
<td>53.32</td>
</tr>
<tr>
<td>P</td>
<td>DCT</td>
<td></td>
<td>41.43</td>
<td>46.27</td>
<td>47.51</td>
</tr>
<tr>
<td></td>
<td>ZTE</td>
<td></td>
<td>42.06</td>
<td>49.05</td>
<td>49.54</td>
</tr>
<tr>
<td></td>
<td>EBCOT</td>
<td></td>
<td>43.53</td>
<td>48.56</td>
<td>49.84</td>
</tr>
<tr>
<td>B</td>
<td>DCT</td>
<td></td>
<td>40.19</td>
<td>45.63</td>
<td>45.85</td>
</tr>
<tr>
<td></td>
<td>ZTE</td>
<td></td>
<td>40.53</td>
<td>46.22</td>
<td>46.69</td>
</tr>
<tr>
<td></td>
<td>EBCOT</td>
<td></td>
<td>40.94</td>
<td>46.54</td>
<td>46.66</td>
</tr>
</tbody>
</table>

Table 4.8: The average PSNRs of *Crowd* sequences for the different type of pictures at 1.00 bpp

<table>
<thead>
<tr>
<th>Frame type</th>
<th>Component Coding</th>
<th>Stream</th>
<th>Main stream (dB)</th>
<th>Auxiliary stream (dB)</th>
<th>avg. bits (kbits)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Y</td>
<td>U</td>
<td>V</td>
</tr>
<tr>
<td>I</td>
<td>DCT</td>
<td></td>
<td>40.79</td>
<td>48.16</td>
<td>49.23</td>
</tr>
<tr>
<td></td>
<td>ZTE</td>
<td></td>
<td>42.05</td>
<td>49.25</td>
<td>50.31</td>
</tr>
<tr>
<td></td>
<td>EBCOT</td>
<td></td>
<td>42.78</td>
<td>50.22</td>
<td>51.02</td>
</tr>
<tr>
<td>P</td>
<td>DCT</td>
<td></td>
<td>39.95</td>
<td>45.26</td>
<td>45.98</td>
</tr>
<tr>
<td></td>
<td>ZTE</td>
<td></td>
<td>40.46</td>
<td>47.93</td>
<td>47.86</td>
</tr>
<tr>
<td></td>
<td>EBCOT</td>
<td></td>
<td>41.44</td>
<td>46.41</td>
<td>47.58</td>
</tr>
<tr>
<td>B</td>
<td>DCT</td>
<td></td>
<td>39.49</td>
<td>44.99</td>
<td>45.05</td>
</tr>
<tr>
<td></td>
<td>ZTE</td>
<td></td>
<td>39.38</td>
<td>46.73</td>
<td>47.58</td>
</tr>
<tr>
<td></td>
<td>EBCOT</td>
<td></td>
<td>39.98</td>
<td>46.43</td>
<td>47.02</td>
</tr>
</tbody>
</table>
By restraining the objective quality of the chrominance components of the reconstructed frames in both main stream and auxiliary stream, the average PSNR of the luminance component of entire sequences can intimate the performance of the stereoscopic video codec at specific bit rate.

The simulation results tabulated in above tables show the efficiency of the proposed codec against the DCT based codec and ZTE based codec. Obviously, as referring the results in luminance component, the proposed codec outperforms the DCT based codec by 1 – 1.5 dB or more and also fulfill the performance over the ZTE based codec by 0.5 – 1 dB depending on the bit rate.

Figure 4.6 and Figure 4.7 display the subjective results of the proposed compression scheme compared with the DCT coding scheme and ZTE coding scheme. As compare the outputs with the DCT based codec, the subjective results are better, in particular, blocking artifacts are reduced, while the decoded views of the ZTE based codec look more indistinct shape than that of the EBCOT based codec. Note that, because of the limit size of papers, the images were cropped to make comparison comfortable for subjective views.
Figure 4.6: The subjective results in frame 30 of *Booksale* sequences (AVDS Sequences) coded at 0.244 bpp. (a) The cropped original right image (b) The cropped right image of DCT coding (c) The cropped right image of ZTE coding (d) The cropped right image of EBCOT coding
Figure 4.7: The subjective results in frame 63 of Crowd sequences (AVDS Sequences) coded at 0.472 bpp. (a) The cropped original right image (b) The cropped right image of DCT coding (c) The cropped right image of ZTE coding (d) The cropped right image of EBCOT coding
4.5 Results of the Effect of the Reference View Selection

So far, the simulations taken place from section 4.1 – 4.4 have used the left view sequence of the stereoscopic video as the reference sequence, or so-called the main stream. Nevertheless, to find out the effect of selecting the reference view for coding, this section is set up to encode by exploiting the right view to be the reference view instead of the left view and compared the quality of reconstructed sequences.

Table 4.9 and Table 4.10 illustrate the simulation results of two sequences at two bit rates with two cases; Left view reference and Right view reference. The system model and system specifications of two cases are similar except the reference view for coding and the disparity scanning range in disparity estimation. Basically, if the left view is used as the reference view and the disparity vectors are defined in positive sign by spanning from the left frame to right frame, when exploiting the right view as the reference, the disparity estimation will scan the matching block in the opposite direction. That is, in the same location of the considering block, the same matching block should be met and get the negative disparity vector. In this simulation, both cases were coded via the EBCOT coding.

Table 4.9: The average PSNRs of Booksale sequences comparing between using Left view and Right view as the main stream at 0.30 bpp

<table>
<thead>
<tr>
<th>Frame type</th>
<th>Stream</th>
<th>Main stream (dB)</th>
<th>Auxiliary stream (dB)</th>
<th>avg. bits (kbits)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Component</td>
<td>Y</td>
<td>U</td>
<td>V</td>
</tr>
<tr>
<td>I</td>
<td>Left view</td>
<td>37.30</td>
<td>46.31</td>
<td>46.95</td>
</tr>
<tr>
<td></td>
<td>Right view</td>
<td>37.23</td>
<td>46.18</td>
<td>47.11</td>
</tr>
<tr>
<td>P</td>
<td>Left view</td>
<td>35.84</td>
<td>41.91</td>
<td>42.31</td>
</tr>
<tr>
<td></td>
<td>Right view</td>
<td>35.92</td>
<td>42.03</td>
<td>42.34</td>
</tr>
<tr>
<td>B</td>
<td>Left view</td>
<td>35.66</td>
<td>40.58</td>
<td>40.84</td>
</tr>
<tr>
<td></td>
<td>Right view</td>
<td>35.62</td>
<td>40.64</td>
<td>40.73</td>
</tr>
</tbody>
</table>

Table 4.10: The average PSNRs of Crowd sequences comparing between using Left view and Right view as the main stream at 0.50 bpp

<table>
<thead>
<tr>
<th>Frame type</th>
<th>Stream</th>
<th>Main stream (dB)</th>
<th>Auxiliary stream (dB)</th>
<th>avg. bits (kbits)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Component</td>
<td>Y</td>
<td>U</td>
<td>V</td>
</tr>
<tr>
<td>I</td>
<td>Left view</td>
<td>40.47</td>
<td>50.58</td>
<td>51.58</td>
</tr>
<tr>
<td></td>
<td>Right view</td>
<td>40.38</td>
<td>50.62</td>
<td>51.47</td>
</tr>
<tr>
<td>P</td>
<td>Left view</td>
<td>38.47</td>
<td>45.71</td>
<td>46.45</td>
</tr>
<tr>
<td></td>
<td>Right view</td>
<td>38.46</td>
<td>45.65</td>
<td>46.42</td>
</tr>
<tr>
<td>B</td>
<td>Left view</td>
<td>37.63</td>
<td>41.96</td>
<td>43.13</td>
</tr>
<tr>
<td></td>
<td>Right view</td>
<td>37.65</td>
<td>42.01</td>
<td>43.23</td>
</tr>
</tbody>
</table>

37
From the simulation results tabulated in above tables, the average PSNRs from simulation by using Left view and Right view as the reference stream are nearly in all types of pictures. By controlling the target bits coded in frame-by-frame of both cases, the slightly different quality ensues due to the following manners:

1. The first reference frames, the I-frame of GOP in main stream, have gotten somewhat diverse quality of intra-frame coding since the details of each frame are not entirely similar.

2. The occluded areas appearing in the auxiliary frames are not lied in the same regions and have the different details comparing between two cases. For the first case, using the left view as the reference, the occlusion takes place in the right side of the foreground. In contrast, the occlusion will locate in the left side of the foreground if the right view is used as the reference.

For this investigation, the selection of the reference stream slightly affects the quality of the reconstructed stereo video sequences. On the other words, the performance of the EBCOT based stereoscopic video codec is steady either the left view or the right view is selected to be the main stream. Nevertheless, according to the scarce subsistence of source sequences used for testing simulation, the general conclusion may not be specifically done. Some video sequences may be recorded in one view which is clearly seen and could be perfectly represented by the second view, while the first view could not absolutely stand for the all regions of the second view due to occlusions.
CHAPTER 5
CONCLUSIONS AND RECOMMENDATIONS

The substantial results, conclusions and the significant supplement of the thesis are summarized in this chapter. The recommendations are also presented so as to give the directions to improve the performance of the proposed scheme.

5.1 Conclusions

This thesis is set to develop a framework for stereoscopic video compression by utilizing the advantages of spatial correlation, temporal correlation and high correlation between stereo image pairs with the aim of increasing the quality of stereo video at the specific bit rate. The EBCOT is considered to be the high effective coding algorithm in the proposed compression scheme as the sharp performance in JPEG2000 standard.

In the proposed scheme, two streams are coded; Main stream and Auxiliary stream. The main stream is coded by DWT and EBCOT algorithm which the prediction runs in a group of pictures as MPEG-2 standard. Since the redundant information between left and right views is quite large, an auxiliary stream can be encoded using the disparity-based compensation prediction. However, sometimes the motion-based compensation provides the better prediction than disparity compensation. The best estimation among these compressions can be achieved. Moreover, fortunately, according to the several perceptual effect experiments, a stereo pair with one sharp image and another blurred image can stimulate appropriate visual perception. Therefore, the quality of images in the auxiliary stream can be slightly lower than that of images in the main stream.

First of all, the performance of the proposed codec has been tested with the gray scale stereo image pairs and compared the performance of EBCOT coding compared with the existing stereoscopic standard codec, MPEG-2, which is based on DCT and one high effective coding, ZTE, applied for stereo images. The significant improvement in PSNR is around 1-3 dB against the DCT coding and around 0.5 - 2 dB against the ZTE coding. Next, the colour stereo image compression has been assessed. To receive further compression efficiency, the colour features of images have been considered to segment the region-of-interest and to predict the occlusion of the stereo image pairs. The performance of the proposed scheme gains 1 - 2 dB compared with the ordinary scheme.

Finally, all proposed schemes have been performed with the stereo video sequences and evaluated the performance against that of DCT based codec and ZTE based codec. By controlling the objective quality of the chrominance components of the reconstructed frames in both main stream and auxiliary stream to have the same values in all codecs, the proposed codec outperforms the DCT based codec by 1 - 1.5 dB or more in Y component and also fulfill the performance over the ZTE based codec by 0.5 - 1 dB in Y component depending on the bit rate. Furthermore, the swapping of the reference stream has been tested and the performance of the EBCOT based stereoscopic video codec shows steady operation either the left view or the right view is selected to be the main stream.
5.2 Recommendations

The recommendations for further study include:

1. The variable size block-based scheme and mesh algorithm should be investigated to improve encoding efficiency along object boundaries.

2. The PSNR difference between main and auxiliary stream can be increased without noticeable distortion whenever the disparity edge is preserved. This may result in increasing compression performance.

3. Other filters, such as Haar filter, to improve transformed performance statistically in residual-frame coding, might substitute the Daubechies 9/7 filter employed in this algorithm.

4. The optimal reference frame can be exploited for best prediction.
REFERENCES


DISTIMA Stereoscopic Image Pairs, <URL: http://www.tnt.uni-hannover.de/project/eu/distima/images/>.


Jeffrey Copper, (2001), <URL: http://www.strabismus.org/all_about_strabismus.html>


Yu Hen Hu, (2001), <URL: http://www.cae.wisc.edu/~ece533/>
APPENDIX A

STEREOSCOPIC VIDEO CODEC ANALYSIS

This appendix explains the analysis of an EBCOT based stereoscopic video codec. It illustrates step-by-step mathematic approaches that were used for programming and simulation in this thesis. Note that the analysis of the stereo image coding can be referred as coding in I-frame in each GOP of the stereo video coding.

A.1 General Definition

The prediction process in this stereoscopic coding runs as Figure 3.2 in the series of \(IBBPBBPBBPBBPB...\) where \(I\) represents I-frame, \(P\) represents P-frame, and \(B\) represents B-frame. For each step of coding excepting the I-frame of main stream coding which is intra-encoded as JPEG2000 standard, two types of picture have been concerned; reference frame and predicted frame. The reference frame is the decoded frame used as the reference for predicting the current frame and the predicted frame is the current frame coded as inter-frame coding, namely I-frame in auxiliary stream, P-frame and B-frame in both streams.

- Reference frame:
  Size: \(M \times N\)
  Values of pixel at row \(i^{th}\) and column \(j^{th}\): \(p^R_i, p^G_i, p^B_i\) (RGB domain)
  \(p^Y_i, p^U_i, p^V_i\) (YUV domain)

- Predicted frame:
  Size: \(M \times N\)
  Values of pixel at row \(i^{th}\) and column \(j^{th}\): \(p'^R_i, p'^G_i, p'^B_i\) (RGB domain)
  \(p'^Y_i, p'^U_i, p'^V_i\) (YUV domain)

A.2 Disparity Estimation

The reference frame is the decoded Left frame of the main stream and the predicted frame is the current Right frame of the auxiliary stream. Each frame is divided into many blocks, \(B_{kl}\) where \(1 \leq k \leq \left\lfloor \frac{M}{m} \right\rfloor, 1 \leq l \leq \left\lfloor \frac{N}{n} \right\rfloor\), with size \(m \times n\).

![Figure A.1: Frame size M x N with blocks size m x n](image)
\[
B_{ij} = \begin{cases} 
  p_{ij}^x \quad & x = Y, U, V \\
  p_{ij}^i \quad & i = (k-1) \cdot m + i; \quad 1 \leq i \leq m \\
  p_{ij}^j \quad & j = (l-1) \cdot n + j; \quad 1 \leq j \leq n \\
  k = \left[ \frac{i}{m} \right], \quad l = \left[ \frac{j}{n} \right] 
\end{cases}
\] (A.1)

In this thesis, two approaches for scanning have been considered.

A.2.1 Uniform Scanning Range

For finding the matching block of the Right frame from the reference frame, the scanning area is defined to search the minimum sum of absolute difference (SAD).

Each scanning area has width \(w_{\text{uni}}\) and height \(h_{\text{uni}}\). The total number of bits used for disparity vectors before coding, \(\hat{b}_{\text{uni}}\), is

\[
\hat{b}_{\text{uni}} = \sum_{k=1}^{M/m} \sum_{l=1}^{N/n} \left( \lceil \log_2 w_{\text{uni}} \rceil + \lceil \log_2 h_{\text{uni}} \rceil \right)
\]

\[
= \frac{M \cdot N}{m \cdot n} \left( \lceil \log_2 w_{\text{uni}} \rceil + \lceil \log_2 h_{\text{uni}} \rceil \right)
\] (A.2)

A.2.2 Variable Scanning Range

By using colour information, the proper scanning ranges can be adjustably defined to each sub-block with size \(m' \times n'\), where \(\frac{m'}{m} = s\) and \(\frac{n'}{n} = t\); \(s\) and \(t\) are positive integers.

Each pixel, \(p_{ij}\), in the chromatic image constitutes of Red colour, \(p_{ij}^R\), Green colour, \(p_{ij}^G\), and Blue colour, \(p_{ij}^B\), components.

Finding the dominant colour, \(R_{ij}\), \(G_{ij}\), \(B_{ij}\), \(U_{ij}\) for Red dominance, Green dominance, Blue dominance and Undefined colour dominance, respectively, of a specific pixel \(p_{ij}\), by initially \(R_{ij} = G_{ij} = B_{ij} = U_{ij} = 0\) and will be equal to 1 if it is dominance.

\[
p_{ij} = \begin{cases} 
  R_{ij} & \text{if } (p_{ij}^R - p_{ij}^G) > \tau_1 \text{ and } (p_{ij}^R - p_{ij}^B) > \tau_1 \\
  G_{ij} & \text{if } (p_{ij}^G - p_{ij}^R) > \tau_1 \text{ and } (p_{ij}^G - p_{ij}^B) > \tau_1 \\
  B_{ij} & \text{if } (p_{ij}^B - p_{ij}^R) > \tau_1 \text{ and } (p_{ij}^B - p_{ij}^G) > \tau_1 \\
  U_{ij} & \text{otherwise}
\end{cases}
\] (A.3)

where \(\tau_1\) is the threshold set for rising up the dominant colour distinctively from others.

It can be marked image regions as containing a diverse dominant colour by eliminating unmarked pixels and other pixels, which are not marked in the same colour. By this approach, the distinct objects could be made clearly noticeable. Nevertheless the next
threshold, $\tau_2$, is defined to make the decision of disparity estimation easier and to avoid ambiguity of colour information. When considering pixels within a sub-block, if

$$
\sum_{j=1}^{n'} \sum_{j'=1}^{n'} R_{ij} > \tau_2,
$$

that sub-block is considered to be Red dominant and named a Red marked sub-block. If

$$
\sum_{j=1}^{n'} \sum_{j'=1}^{n'} G_{ij} > \tau_2
$$

or if

$$
\sum_{j=1}^{n'} \sum_{j'=1}^{n'} B_{ij} > \tau_2,
$$

such sub-blocks are marked Green or Blue, respectively. If neither of the above three conditions are satisfied, a sub-block is named, unmarked.

By using the marked area, at this instant the maximum scanning windows for each sub-block can be defined:

The number of scanning range is $\eta$, which is positive integer. Overhead of each scanning range is $\delta \text{ bits, } \epsilon = 1, 2, \ldots, \eta$. Each scanning area has width $w_\var^\text{var}$ and height $h_\var^\text{var}$. The total number of bits used for disparity vector before coding, $b_\var^\text{dis}^\text{var}$, is

$$
\hat{b}_\var^\text{dis} = \sum_{k=1}^{M} \sum_{i=1}^{N} \left( \left\lfloor \log_2 \delta_{ik} \right\rfloor + \left\lfloor \log_2 w_\var^\text{var} \right\rfloor + \left\lfloor \log_2 h_\var^\text{var} \right\rfloor \right) \quad \text{(A.4)}
$$

For perfect parallel camera configuration,

$$
\hat{b}_\var^\text{dis}^\text{var} = \sum_{k=1}^{M} \sum_{i=1}^{N} \left( \left\lfloor \log_2 \delta_{ik} \right\rfloor + \left\lfloor \log_2 w_\var^\text{var} \right\rfloor \right) \quad \text{(A.5)}
$$

**Criteria:** finding the best matching block that has the minimum $\text{SAD}$ only in the luminance component.

$$
\text{SAD} = \sum_{i=1}^{m} \sum_{j=1}^{n} \left| \tilde{p}_{ij}^\text{y} - p_{(\bar{x}+\bar{y})(\bar{x}+\bar{y})} \right| \quad \text{(A.6)}
$$

where

$$
\bar{y} = (-\left\lfloor \frac{h_{kl}^\text{var}}{2} \right\rfloor + 1), (-\left\lfloor \frac{h_{kl}^\text{var}}{2} \right\rfloor + 2), \ldots, 0, \ldots, \left\lfloor \frac{h_{kl}^\text{var}}{2} \right\rfloor
$$

$$
\bar{x} = 0, 1, \ldots, (w_{kl}^\text{var} - 1)
$$

Because of imperfection of the camera configuration, the other scanning area is defined for achieving better performance,

$$
\tilde{y} = (-\left\lfloor \frac{h_{kl}^\text{var}}{2} \right\rfloor + 1), (-\left\lfloor \frac{h_{kl}^\text{var}}{2} \right\rfloor + 2), \ldots, 0, \ldots, \left\lfloor \frac{h_{kl}^\text{var}}{2} \right\rfloor
$$

$$
\tilde{x} = (-\left\lfloor \frac{w_{kl}^\text{var}}{4} \right\rfloor), \ldots, 0, \ldots, (w_{kl}^\text{var} - 1 - \left\lfloor \frac{w_{kl}^\text{var}}{4} \right\rfloor)
$$

47
\textbf{Matching block:} considering the luminance component, each pixel in the matching block is
\[
\tilde{P}_{ij}^y = \left\{ \begin{aligned}
P_{(i_{u-m+i},j_{v-n+j})}^y \quad & \text{if} \quad (\tilde{x}_{u}, \tilde{y}_{v}) \in (\tilde{x}, \tilde{y}); \\
\sum_{i=1}^{m} \sum_{j=1}^{n} |P_{ij}^y - P_{(i_{u-m+i},j_{v-n+j})}^y|_{(i,j) \in R_u} \quad & = \min \left\{ \sum_{i=1}^{m} \sum_{j=1}^{n} |P_{ij}^y - P_{(i_{u-m+i},j_{v-n+j})}^y|_{(i,j) \in R_v} \right\}
\end{aligned} \right.
\]
\hspace{1cm} (A.7)

A.3 Motion Estimation

The reference frame is the decoded frame in the same stream of the predicted frame and depends on its type as MPEG-2 standard.

A.3.1 Uni-directional Prediction

The P-frame is predicted from the previous I-frame. This method is almost similar to disparity estimation except in the scanning area.

Each scanning area has width \(w_{\text{mot}}\) and height \(h_{\text{mot}}\). The total number of bits used for disparity vectors before coding, \(\hat{b}_{\text{mot}}\), is
\[
\hat{b}_{\text{mot}} = \sum_{k=1}^{M} \sum_{l=1}^{N} \left( \left\lfloor \log_2 w_{\text{mot}} \right\rfloor + \left\lceil \log_2 h_{\text{mot}} \right\rceil \right)
\hspace{1cm} (A.8)
\]
\[
= MN \cdot \frac{M+N}{m \cdot n} \left( \left\lfloor \log_2 w_{\text{mot}} \right\rfloor + \left\lceil \log_2 h_{\text{mot}} \right\rceil \right) \hspace{1cm} (A.9)
\]

Almost, the scanning area is defined equally in horizontal and vertical direction.

\[
\hat{b}_{\text{mot}} = 2 \frac{M \cdot N}{m \cdot n} \left( \left\lfloor \log_2 w_{\text{mot}} \right\rfloor \right) = 2 \frac{M \cdot N}{m \cdot n} \left( \left\lceil \log_2 h_{\text{mot}} \right\rceil \right) \hspace{1cm} (A.10)
\]

A.3.2 Bi-directional Prediction

This approach is different from the previous one in the number of reference frames. This prediction set for B-frame use two reference frames; one previous closest I- or P-frame and one future closest I- or P-frame.

However, this method has drawback in the bit overhead used to identify which reference frame that block belong to. The total number of bits used for disparity vectors before coding, \(\hat{b}_{\text{mot}}\), is
\[
\hat{b}_{\text{mot}} = \frac{M \cdot N}{m \cdot n} (1 + \left\lfloor \log_2 w_{\text{mot}} \right\rfloor + \left\lceil \log_2 h_{\text{mot}} \right\rceil ) \hspace{1cm} (A.10)
\]

Note that the excess bit might be less that one after variable length coding.

In addition, to make the prediction alternatively and precisely, the value of the reference frames for predicting could be proportionally weighed with distance between the current frame and the reference frame.
A.4 Combinational Estimation

This is the combination between disparity estimation and motion estimation as of B-frame in the auxiliary stream. Three reference frames are used as shown in Figure 3.2. Also, this method has drawback of the bit overhead used to identify which reference frame that block belong to.

A.5 Half-pixel Precision

Considering each pixel in the reference frames, in the search process, the eight new positions with a distance of half a pixel around the final integer pixel are tested.

\[
P^Y \left( X_{m+i} \times Y_{n+j} \right) = \left\{ \begin{array}{ll}
\frac{1}{2} \left( P^Y \left( X_{m+i} \times Y_{n+j} \right) + P^Y \left( X_{m+i+1} \times Y_{n+j} \right) \right) \\
\frac{1}{4} \left( P^Y \left( X_{m+i} \times Y_{n+j} \right) + P^Y \left( X_{m+i+1} \times Y_{n+j} \right) + P^Y \left( X_{m+i} \times Y_{n+j+1} \right) + P^Y \left( X_{m+i+1} \times Y_{n+j+1} \right) \right)
\end{array} \right.
\]

The motion vector overhead may be increased by 2 bits per vector; however, in practice due to variable length coding, this might be less than 2 bits. Despite this increase in motion vector overhead, the efficiency of motion compensation outweighs the extra bits, and the overall bit rate is reduced since motion compensated errors are smaller.

Now, consider the luminance component, each pixel in the matching block will be

\[
\hat{P}^Y_{ij} = \left\{ \begin{array}{ll}
\hat{P}^Y \left( X_{m+i} \times Y_{n+j} \right) = \left( \hat{X}_i, \hat{Y}_j \right) \\
\min \left\{ \sum_{i=1}^{m} \sum_{j=1}^{n} \hat{P}^Y_{ij} - P^Y \left( X_{m+i} \times Y_{n+j} \right) \right\} : \alpha, \beta \in \{-1, 0, 1\}
\end{array} \right.
\]

(A.11)
A.6 Overlapped Block Motion/Disparity Compensation

Figure A.3: Sub-block for overlapped block motion/disparity compensation

Using *Raised Cosine* window as a weighing window,

\[
W_{kl} = \frac{1}{4} \left( 1 - \cos \left( \frac{k + \frac{1}{2}}{m} \right) \right) \left( 1 - \cos \left( \frac{i + \frac{1}{2}}{n} \right) \right); \quad \hat{k} = 0, \ldots, (2m-1) \quad \hat{i} = 0, \ldots, (2n-1)
\]

(A.12)

Each pixel of estimated and compensated frame for luminance component is

\[
\hat{p}_{ij}^{y} = \sum_{q=1}^{1} \sum_{r=-1}^{1} (W_{i,j} \cdot p_{i,j}^{y})
\]

(A.13)

\[
\begin{align*}
\xi_{i} &= \hat{i} + \frac{m}{2} - 1 \\
\xi_{j} &= \hat{j} + \frac{n}{2} - 1 \\
\lambda_{1} &= \hat{x}_{(i,q)(j,r)} + i + \frac{\alpha_{(i,q)(j,r)}}{2} \\
\lambda_{2} &= \hat{y}_{(i,q)(j,r)} + n + j + \frac{\beta_{(i,q)(j,r)}}{2}
\end{align*}
\]

*Residual Frame*: the subtracted target frame from the reference frame

Consider the luminance component of each pixel,

\[
\mathcal{R}_{ij}^{y} = p_{ij}^{y} - \hat{p}_{ij}^{y}; \quad 1 \leq i \leq M, 1 \leq j \leq N
\]

(A.14)

So, \( \mathcal{R}_{ij}^{y} \) is signed value.

A.7 Level Offset

If each \( b_{p} \)-bit pixel sample values are unsigned quantities, especially in the case of I-frame of the main stream, an offset of \(-2^{b_{p}-1}\) is added so that the samples have a signed representation in the range
The motivation for the offset is that almost all of the subband samples produced by the DWT involve high-pass filtering and hence have a symmetric distribution about 0. Without the level offset, the LL subband would present an exception, introducing some irregularity into efficient implementations of the standard (Taubman, 2002).

### A.8 Discrete Wavelet Transform

DWT comprises of two major operations; filtering and subsampling. In image coding, DWT is implemented by using appropriately designed quadrature mirror filters (QMFs) or biorthogonal filters. The filter consists of high-pass and low-pass filters. With each application of the filters, the original signal is successively decomposed into components of lower resolution, while the high frequency components are not analyzed any further.

After passing the signal through a half band low-pass filter, half of the samples can be eliminated according to the Nyquist’s rule. Simply discarding every other sample will subsample the signal by two, and the signal will then have half the number of points. Note, however, the subsampling operation after filtering does not affect the resolution, since removing half of the spectral components from the signal makes half the number of samples redundant anyway.

In this thesis the irreversible transformation is exploited and will be describe in the lifting-based DWT implementation of filtering by Daubechies 9/7 filter (Abtonini, 1992)

\[ X(n) \text{ represents each one dimension of } p_{ij} \text{ for intra-frame coding and of } \mathcal{R}_{ij} \text{ for inter-frame coding.} \]

#### A.8.1 Forward Discrete Wavelet Transform

Equation A.15 describes the 4 “lifting” (1 through 4) and the 2 “scaling” steps (5 and 6) and of the one dimension filtering.

\[
\begin{align*}
[\text{step1}] & \quad Y(2n+1) \leftarrow X(2n+1) - (\alpha \times [X(2n) + X(2n+2)]) \\
[\text{step2}] & \quad Y(2n) \leftarrow X(2n) - (\beta \times [Y(2n-1) + Y(2n+1)]) \\
[\text{step3}] & \quad Y(2n+1) \leftarrow Y(2n+1) - (\gamma \times [Y(2n) + Y(2n+2)]) \\
[\text{step4}] & \quad Y(2n) \leftarrow Y(2n) - (\delta \times [X(2n-1) + Y(2n+1)]) \\
[\text{step5}] & \quad Y(2n+1) \leftarrow -(1/K) \times Y(2n+1) \\
[\text{step6}] & \quad Y(2n) \leftarrow K \times Y(2n)
\end{align*}
\]  

(A.15)

where \( K = 1.230 \ 174 \ 105, \ \alpha = -1.586 \ 134 \ 342, \ \beta = -0.052 \ 980 \ 118, \ \gamma = 0.882 \ 911 \ 075 \) and \( \delta = 0.443 \ 506 \ 852 \)

#### A.8.2 Inverse Discrete Wavelet Transform

Equation A.16 describes the 2 “scaling” (1 and 2) and the 4 “lifting” steps (3 through 6) and of the one dimension filtering.
\[
\begin{align*}
\text{[step1]} & \quad X(2n) \leftarrow K \times Y(2n) \\
\text{[step2]} & \quad X(2n+1) \leftarrow -(1/K) \times Y(2n+1) \\
\text{[step3]} & \quad X(2n) \leftarrow X(2n) - (\delta \times [X(2n-1) + X(2n+1)]) \\
\text{[step4]} & \quad X(2n+1) \leftarrow X(2n+1) - (\gamma \times [X(2n) + X(2n+2)]) \\
\text{[step5]} & \quad X(2n) \leftarrow X(2n) - (\beta \times [X(2n-1) + X(2n+1)]) \\
\text{[step6]} & \quad X(2n+1) \leftarrow X(2n+1) - (\alpha \times [X(2n) + X(2n+2)])
\end{align*}
\] (A.16)

where \( K = 1.230 \ 174 \ 105 \), \( \alpha = -1.586 \ 134 \ 342 \), \( \beta = -0.052 \ 980 \ 118 \), \( \gamma = 0.882 \ 911 \ 075 \) and \( \delta = 0.443 \ 506 \ 852 \)

A.9 Quantization

Subband samples, \( y_b[n] \), are mapped to quantization indices, \( q_b[n] \), by using the dead zone scalar quantization.

\[
q_b[n] = \text{sign}(y_b[n]) \left\lfloor \frac{|y_b[n]|}{\Delta_b} \right\rfloor
\] (A.17)

The step size for each subband is specified in terms of an exponent, \( \varepsilon_b \), and a mantissa, \( \mu_b \).

\[
\Delta_b = 2^{\varepsilon_b} \left( 1 + \frac{\mu_b}{2^{|\varepsilon_b|}} \right), \quad 0 \leq \varepsilon_b < 2^i, \quad 0 \leq \mu_b < 2^i
\] (A.18)

A.10 EBCOT Coding

A.10.1 Tier-1

The quantized transform coefficients, \( q_b[n] \), are expressed in a sign-magnitude representation. The significance state changes from insignificant to significant at the bit-plane where the most significant 1 bit is found. In general, a current coefficient can have 256 possible context vectors. These are clustered into a smaller number of contexts according to the rules specified below for context formation. Four different context formation rules are defined, one for each of the four coding operations: significance coding, sign coding, magnitude refinement coding, and cleanup coding. These coding operations are performed in three coding passes over each bit plane: significance and sign coding in a significance propagation pass, magnitude refinement coding in a magnitude refinement pass, and cleanup and sign coding in a cleanup pass. For further details, the reader is referred to Boliek (2000).

Therefore, the output of the Tier-1 encoder, embedded code, comprises of three coding passes per bit plane, that is:

i) Significance Pass

This pass is used to convey significance and sign information for samples. The samples in the code-block are scanned in the order shown in Figure A.4.
Membership in this pass can be expressed by the conditions
\[
\sigma(x) = 0 \text{ and } \sigma(h) + \sigma(v) + \sigma(d) > 0
\]  
(A.19)
where \(\sigma()\) shows the significance of the pixel, if the pixel is significant then \(\sigma() = 1\) and if the pixel is insignificant then \(\sigma() = 0\). As well as \(x\) represents the current bit, \(h\) represents the horizontal neighbors of \(x\), \(v\) represents the vertical neighbors of \(x\) and the \(d\) represents diagonal neighbors of \(x\). Position of \(x, h, v\) and \(d\) are shown in Figure A.5.

Algorithm 1 describes the significance pass (Adams, 2002).

### Algorithm 1

1: for each sample in code-block do
2:     if sample previously insignificant and predicted to become significant during current bit plane then
3:         code significance of sample /* 1 binary symbol */
4:     if sample significant then
5:         code sign of sample /* 1 binary symbol */
6:     endif
7:  endif
8: endfor

\section*{ii) Refinement Pass}

This pass signals subsequent bits after the most significant bit for each sample. The samples of the code-block are scanned using the order shown earlier in Figure A.4. If a sample was found to be significant in a previous bit plane, the next most significant bit of that sample is conveyed using a single binary symbol. Algorithm 2 describes this process in pseudo code form (Adams, 2002).
iii) Cleanup Pass

This pass is used for scanning remaining bits in bit plane (remaining from significance pass and refinement pass). The cleanup pass is not much different from the significance pass. The key difference is that the cleanup pass conveys information about samples that are predicted to remain insignificant, rather than those that are predicted to become significant. Membership in this pass can be expressed by the conditions

\[ \sigma(x) = 0 \text{ and } \sigma(h) + \sigma(v) + \sigma(d) = 0 \]  
\[ (A.20) \]

Algorithm 3 describes this process in pseudo code form (Adams, 2002).

---

Algorithm 2 Refinement pass algorithm.

1: for each sample in code-block do
2:   if sample found significant in previous bit plane then
3:     code next most significant bit in sample /* 1 binary symbol */
4:   endif
5: endfor

Algorithm 3 Cleanup pass algorithm.

1: for each vertical scan in code-block do
2:   if four samples in vertical scan and all previously insignificant and unvisited
   and none have significant 8-connected neighbor then
3:     code number of leading insignificant samples via aggregation
4:     skip over any samples indicated as insignificant by aggregation
5:   endif
6: while more samples to process in vertical scan do
7:   if sample previously insignificant and unvisited then
8:     code significance of sample if not already implied by run /* 1 binary symbol */
9:     if sample significant then
10:    code sign of sample /* 1 binary symbol */
11:   endif
12: endif
13: endwhile
14: endfor

After each bitplane is fractioned to be significance passes, refinement passes and cleanup passes then each pass is converted to contexts and symbols prepared for arithmetic coder. The algorithms for coding these contexts can be divided into four coding: run length coding (RLC), zero coding (ZC), magnitude refinement (MR) and sign coding (SC). After that the obtained contexts and symbols of each pass are compressed to obtain the compressed passes by using arithmetic coder.

- Run Length Coding (RLC)

The RLC is used for reducing the average number of binary symbols. The bits are coded by RLC that must satisfy the following conditions:

1) Four bits in same stripe must all be insignificant that can be expressed by
\[ \sigma(k1, k2 + z) = 0 \quad ; \quad z \in \{0, 1, 2, 3\} \quad (A.21) \]

where \(k1\) is position on horizontal axis and \(k2\) is position on vertical axis. And \(k2\) is first bit in a stripe or \(k2 \in \{0, 4, 8, \ldots \}\) corresponding to Figure A.4

2) The bits must have insignificant neighbors that can be expressed by

\[ \sigma(k1 \pm 1, k2 + z) = 0 \quad ; \quad z \in \{0, 1, 2, 3\} \quad \text{and} \quad (A.22) \]
\[ \sigma(k1 \pm 1, k2 + z) = 0 \quad ; \quad y \in \{-1, 0, 1\}, \quad z \in \{-1, 4\} \quad (A.23) \]

If a group of four samples satisfies the above conditions and all four bits are 0 then these four bits are coded with context 00H and symbol 0. And if a group of four samples satisfies the above conditions and at least one bit are 1 then position of first bit 1 is coded by using two contexts 12H and two symbols corresponding to Table A.1. And if following bits are bit 0s then they are code with ZC. If following bits are coded with ZC. This coding is only used in cleanup pass.

<table>
<thead>
<tr>
<th>Bits allocation</th>
<th>Context</th>
<th>Symbol 1, 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 x x x</td>
<td>12H</td>
<td>00</td>
</tr>
<tr>
<td>0 1 x x</td>
<td>12H</td>
<td>01</td>
</tr>
<tr>
<td>0 0 1 x</td>
<td>12H</td>
<td>10</td>
</tr>
<tr>
<td>0 0 0 1</td>
<td>12H</td>
<td>11</td>
</tr>
</tbody>
</table>

• **Zero Coding (ZC)**

The ZC is used for coding the bits that are neighbors of significant pixel and used in RLC. Almost all the relevant information appears to be captured by the significance of these neighbors, which group into the following three categories:

1) Horizontal: \(h_i(k) = \sum_{z \in \{-1,1\}} \sigma(k1 + z, k2)\) so that \(0 \leq h_i(k) \leq 2\)

2) Vertical: \(v_i(k) = \sum_{z \in \{-1,1\}} \sigma(k1, k2 + z)\) so that \(0 \leq v_i(k) \leq 2\)

3) Diagonal: \(d_i(k) = \sum_{z1, z2 \in \{-1,1\}} \sigma(k1 + z1, k2 + z2)\) so that \(0 \leq d_i(k) \leq 4\)

Neighbors that lie outside the code-block are interpreted as insignificant, so as to ensure that the block bit streams are truly independent. No such assumption is imposed on neighbors that lie outside the relevant sub block; however, sub blocks are by no means independent. To minimize both model adaptation cost and implementation complexity, the 256 possible neighborhood configurations are quantized to nine distinct coding contexts, with the labels indicated in Table A.2. And the symbol depends on the value of current bit. For an example, if current bit = 1 then symbol = 1, if current bit = 0 then symbol = 0.

• **Magnitude Refinement (MR)**

The MR is used to refine the bits that their pixel is already significant. It has three contexts that depend upon all its neighbors and bits in the same pixel that higher bit plane
than current bit. The contexts are defined as Table A.3 and the symbol depends on the value of current bit. For an example, if current bit = 1 then symbol = 1, if current bit = 0 then symbol = 0.

Table A.2: Assignment of nine ZC contexts (Boliek, 2000)

<table>
<thead>
<tr>
<th>LH and LL Sub-band (vertically high pass)</th>
<th>HL Sub-band (horizontally high pass)</th>
<th>HH Sub-band (diagonally high pass)</th>
</tr>
</thead>
<tbody>
<tr>
<td>h</td>
<td>v</td>
<td>d</td>
</tr>
<tr>
<td>2</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>1</td>
<td>≥1</td>
<td>x</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>≥1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>2</td>
<td>x</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>x</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>≥2</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Table A.3: Assignment of three MR contexts (Boliek, 2000)

<table>
<thead>
<tr>
<th>σ(m,n)</th>
<th>h_i(k) + v_i(k) + d_i(k)</th>
<th>Context</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>x</td>
<td>0CH</td>
</tr>
<tr>
<td>0</td>
<td>≥1</td>
<td>0BH</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0AH</td>
</tr>
</tbody>
</table>

where $\sigma(m,n)$ is second state variable. If the number of bit 1s of the higher bitplane in the same pixel (more significant) > 1 then $\sigma(m,n) = 1$. If the successive higher bit in the same pixel (more significant) = bit 1 and the number of bit 1s of the higher bitplane in the same pixel (more significant) =1 then $\sigma(m,n) = 0$.

- **Sign Coding (SC)**

The SC primitive is used at most once for each sample, immediately after a previously insignificant sample is first found to be significant during a ZC or RLC operation. It has five contexts that depend upon horizontal neighbors and vertical neighbors. The contexts are defined as Table A.4. And the symbol is 0 for all time. $h_i(k)$ and $v_i(k)$ are horizontal contribution and vertical contribution, respectively. If one or both horizontal neighbors are negative, $h_i(k) = -1$. If both horizontal neighbors are insignificant or both horizontal neighbors are significant but opposite sign, $h_i(k) = 0$. And if one or both horizontal neighbors are positive, $h_i(k) = 1$. For vertical, if one or both vertical neighbors are negative, $v_i(k) = -1$. If both vertical neighbors are insignificant or both vertical
neighbors are significant but opposite sign, \( \overline{v}_v(k) = 0 \). And if one or both vertical neighbors are positive, \( \overline{v}_v(k) = 1 \).

From Table A.4, this context is provided for the arithmetic decoder with the bit stream. The bit returned is then logically exclusive ORed with the XORbit to produce the sign bit. The following equation is used

\[
\text{signbit} = \text{AC(context)} \oplus \text{XORbit}
\]

where signbit is the sign of the current coefficient (bit 1 indicates a negative coefficient, bit 0 indicates a positive coefficient), AC(context) is the value returned from the arithmetic decoder given the context and the bit stream and XORbit is found in Table A.4 for the current context.

Table A.4: Assignment of three SC contexts (Boliek M., 2000)

<table>
<thead>
<tr>
<th>( \overline{h}_v(k) )</th>
<th>( \overline{v}_v(k) )</th>
<th>Context</th>
<th>XORbit</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>11H</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>10H</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>-1</td>
<td>0FH</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0EH</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0DH</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>-1</td>
<td>0EH</td>
<td>1</td>
</tr>
<tr>
<td>-1</td>
<td>1</td>
<td>0FH</td>
<td>1</td>
</tr>
<tr>
<td>-1</td>
<td>0</td>
<td>10H</td>
<td>1</td>
</tr>
<tr>
<td>-1</td>
<td>-1</td>
<td>11H</td>
<td>1</td>
</tr>
</tbody>
</table>

In conclusion, the Tier-1 encoder generates a separate highly scalable (or embedded) bit-stream for each code-block.

Although, there are as many coefficients as there are samples, the information content tends to be concentrated in just a few coefficients. Through quantization, the information content of a large number of small-magnitude coefficients is further reduced. Additional processing by the entropy coder reduces the number of bits required to represent these quantized coefficients, sometimes significantly compared to the original image. For the arithmetic entropy coding, all of steps are clearly described in Boliek (2000) in Annex C.

A.10.2 Tier-2

For the Tier-2 encoder, the coding pass information is packaged into data units called packets comprising of two parts: a header and body, in a process referred to as packetization. The header indicates which coding passes are included in the packet, while the body contains the actual coding pass data itself. One packet is generated for each component, resolution level, layer, and precinct 4-tuple which is a grouping of code-blocks within a sub-band. To take advantage of this, there are two coding in this stage:
i) Packet Header Coding

Algorithm 4 describes the Packet Header Coding (Adams, 2002).

Algorithm 4 Packet header coding algorithm.
1: \textbf{if} packet not empty \textbf{then}
2: \hspace{1em} code non-empty packet indicator /* 1 binary symbol */
3: \hspace{1em} \textbf{for} each sub-band in resolution level \textbf{do}
4: \hspace{2em} \textbf{for} each code-block in sub-band precinct \textbf{do}
5: \hspace{3em} code inclusion information /* 1 binary symbol or tag tree */
6: \hspace{2em} \textbf{if} no new coding passes included \textbf{then}
7: \hspace{3em} \hspace{1em} skip to next code-block
8: \hspace{2em} \textbf{endif}
9: \hspace{2em} \textbf{if} first inclusion of code-block \textbf{then}
10: \hspace{3em} code number of leading insignificant bit planes /* tag tree */
11: \hspace{2em} \textbf{endif}
12: \hspace{2em} code number of new coding passes
13: \hspace{2em} code length increment indicator
14: \hspace{2em} code length of coding pass data
15: \hspace{2em} \textbf{endfor}
16: \hspace{1em} \textbf{endfor}
17: \textbf{else}
18: \hspace{1em} code empty packet indicator /* 1 binary symbol */
19: \hspace{1em} \textbf{endif}
20: pad to byte boundary

Algorithm 5 Packet body coding algorithm.
1: \textbf{for} each sub-band in resolution level \textbf{do}
2: \hspace{1em} \textbf{for} each code-block in sub-band precinct \textbf{do}
3: \hspace{2em} \textbf{if} (new coding passes included for code-block) \textbf{then}
4: \hspace{3em} output coding pass data
5: \hspace{2em} \textbf{endif}
6: \hspace{2em} \textbf{endfor}
7: \hspace{1em} \textbf{endfor}

ii) Packet Body Coding

Algorithm 5 describes the Packet Header Coding (Adams, 2002).

The bit stream data created from these coding passes is conceptually grouped in layers. Layers are an arbitrary number of groupings of coding passes from each code-block. Although there is great flexibility in layering, the basic premise is that each successive layer contributes to a higher quality image.

Packets are a fundamental unit of the compressed codestream. A packet is a particular partition of one layer of one decomposition level of one tile-component. This partition provides another method for extracting a spatial region independently from the codestream. These packets may be interleaved in the codestream using a few different methods.
A.11 Rate Control

Either varying the quantization step sizes or the selection of the subset of coding passes to include in the code stream can achieve rate control in the encoder. For the first approach, the rate decreases as the step sizes are increased and make the result that distortion is increased. For the second approach, the encoder can elect to discard coding passes in order to achieve the specific rate. According to the capability of calculating the distortion reduction, the encoder can include this information in order to decrease distortion reduction per unit until the bit budget has been exhausted.

Each component of the wavelet transformed image is partitioned into a collection of relatively small code-blocks, $B_i$, whose entropy-coded embedded bit-streams may be truncated to rates, $R_i$, according to the specified maximum bit rate for the final coding stream. The contribution from $B_i$ to distortion in the reconstructed image is denoted $D_i$, for each truncation point, $n_i$. Here the relevant distortion metric can be assumed to be additive, i.e.,

$$D = \sum_{n_i} D_i$$ \hspace{1cm} (A.25)

where $D$ represents the overall image distortion and $n_i$ denotes the truncation point selected for code-block $B_i$. Usually, an additive distortion metric which approximates Mean Squared Error (MSE) is used by setting

$$\hat{D}_i^n = w_{i,b}^n \sum_{k \in B_i} (\hat{s}_i^n[k] - s_i[k])^2$$ \hspace{1cm} (A.26)

Here $s_i[k]$ denotes the 2D sequences of sub-band samples in code-block $B_i$, $\hat{s}_i^n[k]$ denotes the quantized representation of these samples associated with truncation point $n_i$, and $w_{i,b}$ denotes the coefficient weighting for the sub-band, $b_i$, to which code-block $B_i$ belongs.

To minimize distortion subject to a specified maximum bit rate, $R^{\text{max}}$, the rate allocation algorithm must find the optimal set of truncation points that satisfies the following equation:

$$R^{\text{max}} \geq R = \sum_i R_i^{n_i}$$ \hspace{1cm} (A.27)

The problem of selecting the optimal truncation points to satisfy Equation A.27 can be solved using a generalized Lagrange multiplier method (Everett, 1963). Any set of truncation points, $n_i^*$, which minimizes Equation A.28

$$(D(\lambda) + \lambda R(\lambda)) = \sum_i (D_i^{n_i} + \lambda R_i^{n_i})$$ \hspace{1cm} (A.28)

for some $\lambda$ is optimal in the sense that the distortion cannot be reduced without also increasing the overall rate and vice versa.
The determination of the optimal truncation points, \( n^i_k \), for any given \( \lambda \), may be performed very efficiently based on a small amount of summary information collected during the generation of each code-block’s embedded bit-stream. It is clear that, for each code-block \( B_i \), it is a separate minimization problem. A simple algorithm to find the truncation point, \( n^i_k \), which minimizes \((D^i_k + \lambda R^i_k)\), is as follows:

- Initialize \( n^i_k = 0 \);
- For \( j = 1, 2, 3, \ldots \)
  - set \( \Delta R^i_j = R^i_j - R^i_k \) and \( \Delta D^i_j = D^i_k - D^i_j \);
  - If \( \Delta D^i_j / \Delta R^i_j > \lambda \) then update \( n^i_k = j \);

First finding the subset of feasible truncation points, \( N_i \). Let \( j_1 < j_2 < \ldots \) be the numeration of these feasible truncation points and let the corresponding distortion-rate slopes be given by \( S^i_{j_k} = \Delta D^i_j / \Delta R^i_j \). These slopes must be decreasing, for if \( S^i_{j_k} > S^i_{j_l} \) then the truncation point, \( j_k \), could never be selected by the above algorithm, regardless of the value of \( \lambda \), contradicting the fact that \( N_i \) is the feasible set of truncation points.

When restricted to a set of truncation points whose slopes are strictly decreasing, the above algorithm reduces to the trivial selection \( n^i_k = \max \{ j_k \in N_i \mid S^i_{j_k} > \lambda \} \) so that each such point must be a valid candidate for some value of \( \lambda \). It follows that \( N_i \) is the largest set of truncation points for which the corresponding distortion-rate slopes are strictly decreasing. This unique set may be determined using a conventional convex hull analysis (Sedgewick, 1992).

In a typical implementation of the EBCOT algorithm, \( N_i \), is determined instantly after the bit-stream for \( B_i \) has been generated. The rates, \( R^i_k \), and slopes, \( S^i_k \), for each \( j_k \in N_i \), are kept in a compact form along with the embedded bit-stream until all code-blocks have been compressed, at which point the search for the optimal \( \lambda \) and \( n^i_k \) proceeds in a straightforward manner. It is worth emphasizing that only rate and slope values must be stored, not the distortion. This requires only a fraction of the storage for the embedded bit-stream itself (Taubman, 2000).
APPENDIX B

VARIABLE SCANNING RANGE FOR DISPARITY ESTIMATION

This appendix discusses about the variable scanning range for disparity estimation used for stereo image pair and could be applied to stereo video sequences. This proposed algorithm expends the usefulness of chromatic information of both image pairs so as to predefine the spanning window size for searching the corresponding blocks.

B.1 Chromatic Information and Disparity Estimation

In actual fact, colour images compose of three primary colour components; Red, Green and Blue. The remaining colours in a given colour domain can be generated with different weighted combinations of these primary components. Figure B.1 illustrates the Baboon colour image, its luminance component and each of its colour components. A closer inspection of these images indicates that, only using luminance information may not be always sufficient to distinguish between distinct object parts of the image. On the other hand it is seen that the chromatic information could be used to assist the separation of these objects. Further, suppose the high red value areas are of a particular interest to a given application. With chrominance based disparity estimation, it could accurately identify and separate the Baboon’s nose and eyes and use this result in a special coding scheme that protects such regions-of-interest for adverse compression effects.

![Baboon Image Components](image)

(a) Original image (b) Luminance component or Y component (c) Red component (d) Green component and (e) Blue component

**Figure B.1: Comparing components of the Baboon image**

Evidently, stereo image pairs are mostly generated using two identical cameras accurately calibrated so that their image axes are parallel. Under such a geometrical configuration, disparity vectors between the left and the right images do not span
considerably in the perpendicular direction, but are more severe in horizontal direction. Furthermore, the magnitude of disparity vectors depends upon the stereoscopic depth of 3D objects in the stereo scene. 3D object at higher stereoscopic depth would indicate higher magnitude disparity vectors, particularly in the horizontal direction. Hence, to improve disparity estimation, separating the stereoscopic scene into a number of areas is proposed, corresponding to their profundity and designate each area a maximum searching range for estimating correspondence more accurately and efficiently.

One approach to the above disparity estimation technique is to use the chromatic information to classify a stereoscopic scene into areas-of-special-interest covering distinct 3D objects in the scene and plane areas containing minimal stereoscopic data. The following section describes the above approach in detail.

B.2 Proposed Scheme

B.2.1 Definitions

Assume the left image to be the reference image and the right image to be the predicted image. Both the reference image and the predicted image of size $N \times M$ pixels, are divided into non-overlapping sub-blocks of size $m \times n$.

Each pixel, $p_{ij}$ ($0 \leq i \leq M$ and $0 \leq j \leq N$), in the chromatic image constitutes of Red colour, $p_{ij}^R$, Green colour, $p_{ij}^G$, and Blue colour, $p_{ij}^B$, components. Moreover, it can define the dominant colour of a specific pixel $p_{ij}$, for example as Red, $R_{ij}$, if $(p_{ij}^R - p_{ij}^G > \tau_1)$ and $(p_{ij}^R - p_{ij}^B > \tau_1)$, where $\tau_1 > 0$ is a threshold value. Similarly, if $(p_{ij}^G - p_{ij}^R > \tau_1)$ and $(p_{ij}^G - p_{ij}^B > \tau_1)$ or if $(p_{ij}^B - p_{ij}^R > \tau_1)$ and $(p_{ij}^B - p_{ij}^G > \tau_1)$, the dominant colour is $G_{ij}$ or $B_{ij}$, respectively. It is clear that in a particular pixel, $p_{ij}$ can have only one dominant colour, or would not correspond to any dominant colour, $U_{ij}$. If $p_{ij}$ contains a dominant colour, it will be named a marked pixel; else, it will be named as an unmarked pixel.

Figure B.2: The blue dominant object by removing unmarked pixels and pixels marked with other colours (a) Original image (b) Identified blue dominant object

Furthermore, the image regions can be marked as containing a dominant colour by eliminating unmarked pixels and other pixels, which are not marked in the same colour. By
this approach, the distinct objects could be made clearly noticeable (An example is shown in Figure B.2). Nevertheless, the next threshold, $\tau_2$, is defined to make the decision of disparity estimation easier and to avoid ambiguity of colour information. When considering pixels within a sub-block, if $\sum_{i=1}^{m} \sum_{j=1}^{n} R_{ij} > \tau_2$, that sub-block is considered to be Red dominant, the sub-block is named a Red marked sub-block. If $\sum_{i=1}^{m} \sum_{j=1}^{n} G_{ij} > \tau_2$ or if $\sum_{i=1}^{m} \sum_{j=1}^{n} B_{ij} > \tau_2$, such sub-blocks are marked Green or Blue, respectively. If neither of the above three conditions are satisfied, a sub-block is named, unmarked.

**B.2.2 Variable Scanning Range of Disparity Vectors**

In block matching algorithms, the most similar blocks are searched to generate the predicted image. Both the reference picture and the target picture are initially separated into blocks of size $m \times n$ that satisfy the condition $m < m'$ and $n < n'$.

The general uniform scanning block algorithms suffer by the limitation that they use the maximum horizontal and vertical scanning dimensions, $d_{\text{max},x}$ and $d_{\text{max},y}$ respectively, in the disparity estimation of all regions, regardless of the actual stereoscopic depth represented by such regions. In such algorithms, if $\hat{x}$ represents the number of bits used for horizontal disparity vector and $\hat{y}$ represents the number of bits used for the vertical disparity vector, it can be defined as

$$\hat{x} = \lceil \log_2 d_{\text{max},x} \rceil \quad \text{(B.1)}$$

$$\hat{y} = \lceil \log_2 d_{\text{max},y} \rceil \quad \text{(B.2)}$$

where $\lceil x \rceil$ means the nearest integer number that is greater than $x$. Hence, the total number of bits required to represent disparity information can be calculated from Equation B.3.

$$\text{Bit}_{\text{total}} = MN (\hat{x} + \hat{y}) / mn \quad \text{(B.3)}$$

In order to reduce the above bit budget requirement for the total disparity vector field, using variable scanning dimensions for separate regions is considered, particularly in the horizontal direction. As the vertical disparity values are low and do not change considerably within an image, the above-mentioned fixed maximum vertical dimension to code the vertical disparity vectors is exploited. Assume using $\eta$ varieties of maximum horizontal scanning dimensions, the number of bits used for each can be denoted as $\hat{h}_1, \ldots, \hat{h}_\eta$. The quantity of bits used to identify, $\hat{h}_z$, can be found as follows.

$$\hat{h}_z \leq \lceil \log_2 \eta \rceil \quad \text{(B.4)}$$

Therefore, the total number of disparity bits will be,

$$\text{Bit}_{\text{total}} = \sum_{z=1}^{\eta} \alpha_z (\hat{h}_z + \hat{x} + \hat{y}) \quad \text{(B.5)}$$

where $\alpha_z$ is the number of blocks which use $\hat{x}_z$. It can be noticed that $\text{Bit}_{\text{total}}$ used in the variable scanning range algorithm can be more than that of the uniform scanning block algorithm. This drawback can be compensated by various approaches, for instance, since the vertical disparity compared to the horizontal disparity is significantly low, $\hat{y}$ could be
reduced, or since the chromatic component contains less energy compared to the luminance component, the chromatic component can be compressed to a greater extent. The second approach is preferred and can be done by calculating the compression rate of the colour components of the image. The increase in compression rate of the chrominance component is limited by the extra amount of bits needed when variable scanning range algorithm is used as against the uniform scanning range method. Generally, the bits saved by the increase in compression will be greater than the excess amount of bits needed when using uniform scanning, whilst the quality of coded image is maintained at better level.

B.2.3 Disparity Estimation Using Chromatic Information

Firstly, the reference picture and the target picture are scanned to find dominant chromatic regions. Initially colour marked pixels \( R_{ij}, G_{ij} \) and \( B_{ij} \) described above are searched by using the threshold \( \tau_1 \). Subsequently using the threshold \( \tau_2 \), colour marked sub-blocks are constructed. By removing all unmarked pixel sub-blocks belonging to other colours, the regions-of-interest in each colour dominance is separated. From Figure B.3, it is seen that the identified object at the top of the scene needs more horizontal scanning range as compared to that needed by the middle object (i.e. the fish). This is due to the difference in binocular depths of the two objects. The green and red dominant components of the same stereo image pair also produces distinctively identifiable objects, at different positions. Note that the size of sub-blocks \( (m' \times n') \), \( \tau_1 \) and \( \tau_2 \) are chosen large enough to make the decision of disparity estimation easier and to avoid ambiguity of colour information.

![Figure B.3: Correspondence between blue dominant components of the stereo image pair](image)

Secondly, disparity estimation is applied to each colour dominant picture to identify \( \hat{z}_x \) of each region-of-interest. Afterward, in this thesis, the disparity estimation with half-pixel precision is performed, and further supplemented by the well known overlapped block disparity compensation (OBDC) technique. Since, the above block disparity values are computed for fixed-size blocks rather than for pixels or arbitrary shaped features, OBDC could be effectively used to reduce the resulting blocking artifacts. OBDC is known to improve smoothness in disparity vector fields. Before compression, the images are converted to their \( YC_rC_b \) format. Finally, EBCOT algorithm is employed to compress images at desired bit rates, by weighing compression among \( Y \), \( C_r \) and \( C_b \) components. The further details, the reader is referred to Anantrasirichai (2003d) for image coding and Anantrasirichai (2003a) for stereo video coding.
APPENDIX C
OCCLUDED PREDICTION ANALYSIS

The analysis of the occluded prediction for stereo images is illustrated in this appendix. The step-by-step mathematic algorithms and hypothetical rationales are discussed additionally. The advantage of this scheme might be adapted suitably to stereoscopic video sequences.

C.1 Predicting the Occlusion

As using the reconstructed Left image to be the reference image, such as the picture shown in Figure C.1, the occlusions appearing in the Right image, such as the picture shown in Figure C.2, are the regions that cannot be seen in the Left image pair. These might cause the erroneous matching. Therefore, the occluded information is consistently helpful in stereo image coding.

Figure C.1: Reconstructed Left image of Claude4-3 with 1.44 bpp (PSNR = 40.94 dB)

Figure C.2: The original Right image of Claude4-4 that would be coded with 1.2 bpp

To practicable implement, the block-based prediction is exploited as the following.

C.1.1 Edge Detection

To find edges of the image, various methods can be exploited. In this thesis, Canny method has been chosen because of good detection, good localization, low-responses multiplicity of the response to a single step edge (Didier Demigny and Tawfik Kamlé, 1997)

Canny method searches edges of the images by looking for local greatest of the gradient of intensity of image. The gradient is calculated using the derivative of a Gaussian filter. The method uses two sensitivity thresholds, to detect strong and weak edges, and includes the weak edges in the output only if they are connected to strong edges. This
method is therefore less likely than the others to be "fooled" by noise, and more likely to
detect true weak edges. Figure C.3 illustrates the edges of Figure C.2 detected by Canny
method.

Figure C.3: Detected edges of the Right image used Canny method with threshold 0.3
for weak edges and 4 for strong edges

C.1.2 Disparity Estimation

With parameter $r$, the average width of occlusion, depending upon camera
configuration, the occluded compasses in the Right image are expected to occur in the right
side of edges. The parameter $r$ can be identified by calculating the difference between the
amount of disparity vectors of the significant objects in the left side and right side of edges.

\[
r = \frac{1}{R} \sum_{p=1}^{R} r_p
\]

\[
= \frac{1}{R} \sum_{p=1}^{R} \left[ d_{p}^{\text{object}} - d_{p}^{\text{right}} \right]
\]

\[
= \frac{1}{R} \sum_{p=1}^{R} \left[ d_{p}(i, j) - d_{p}(i + B_{\text{size}}, j) \right]
\]

where $d_{p}^{\text{object}}$ is the disparity of the most right pixel of interested object, $p$, locating at $(i,j)$,
while $d_{p}^{\text{right}}$ is the disparity of the pixel locating at next $B_{\text{size}}$ pixels in the right side of $p$. $B_{\text{size}}$
is the horizontal Block size and $R$ is the total number of significant objects. Figure C.4
illustrates the probable region of occlusion with block-based scheme.

Figure C.4: Estimated occluded region in the Right image by Edge detection method
(Block size = 16x16, $r = 3$)

However, only using the edge detection cannot correctly identify the actual
occluded area, since the depth information still be not exploited. Therefore, disparity
vectors are functionally introduced to predict the occluded blocks.
The disparity vector, $d_{kl}$, is the vector from a current considering block in row $k$ and column $l$ of the Right image to the most matching block in the decoded Left image. In this thesis, SAD is exploited to be the criteria for seeking the most matching block.

$$d_{kl} = \left\{ \begin{array}{l} (\bar{x}, \bar{y}) \quad \text{min} \left\{ \sum_{i=1}^{m} \sum_{j=1}^{n} \left| p_{yk}^{r}(x_{i},y_{j}) - p_{yk}^{l}(x_{i},y_{j}) \right| \right\} \\ (\bar{x}, \bar{y}) \in \text{Searching Windows, } W \end{array} \right\}$$

(C.2)

With the perfect parallel-axes camera configuration, the vertical component, $\bar{y}$, could be supposed to be zero and disparity vector remains only the horizontal component, so $d_{kl} = (\bar{x}, \bar{y}) = \bar{x}$.

Obviously, regard to the Left image as the reference, the direction from the Left image to the Right image is considered as the positive direction, the disparity vectors in a stereo image pair are always non-positive. Furthermore, the use of the parallel axes geometry leads to a simple mathematical relationship between the disparity of a point pair and the distance (depth information) to the object it represents.

To prove this, consider the cameras are located on the same X-Y plane of the world coordinate as shown in Figure C.5. The distance between the two cameras is called the baseline distance, denoted by $B$. When $B$ is close to the separation between the two eyes of a person (2.5-3 inches), this configuration simulates the human binocular imaging system.

Figure C.5: Parallel camera configuration (a) 3D view (b) the X-Z plane view
(adapted from Wang et al., 2001)

$C_l$, $C_r$, $C_w$ denote the left, right and world coordinate. The world coordinates of a 3D point $X = [X, Y, Z]^T$, the left and right camera coordinates are related to the left and right image coordinates, $x_l = [x_l, y_l]^T$ and $x_r = [x_r, y_r]^T$ as follows (Wang et al., 2001):
In the above equations, the origin of the world coordinate is set halfway between the two cameras and it is assumed that the two cameras have the same focal lengths, $F$. According to the preceding relation, it should be noticeable that the disparity vector has only a horizontal component and is related to the depth by

$$d = \tilde{x} = x_r - x_l = \frac{-FB}{Z}$$ \hfill (C.5)

Moreover, it can be noticed that the disparity $d$ is always non positive as well as the closer an object to the cameras, the larger the disparity value. On the other words, the magnitude of the disparity $d$ can be used for calculating the depth of the objects in the 3D image.

Hence, the disparity estimation scheme is applied to the likely occluded region of the Right image, to identify the occlusion more precisely. However, for comfortable implementation, the block-based estimation is used and may cause the erroneous occluded regions that occur from the inaccuracy of the estimated disparity vector since all pixels in a block do not share the same disparity parameter. Figure C.6 illustrates the predicting occluded blocks.

**Figure C.6: The shade blocks identify the predicting occluded region**

Subsequently, the block size of occluded region has to reshape in order to be suitable for the independent code block of EBCOT coding. Note that the smallest code block size is 32x32 in EBCOT algorithm. Fortunately, the last estimating step has made the prediction more precise as shown in Figure C.7.

**Figure C.7: The final occluded estimation prepared appropriately for EBCOT algorithm**
C.2 EBCOT Coding for Occlusion

After the occlusion prediction has been exploited, the occluded regions are marked and independently coding separated from the other areas. Basically, one side view of the stereo image pair is predicted from another view used as the reference image to make more efficient compression, and then the residual of the difference between the actual values and the predicted values of this image is coded. To improve the stereo image coding, the occluded regions are coded in parallel and competed with the coded residual block in same location by using the variance of the value of each pixel in that block as the criteria.

Fortunately, the separated coding in some regions is possible for EBCOT algorithm. Methodically, after wavelet transformation and quantization, the pixel transformed coefficients are grouped with rectangular size in power of two, minimum at 32x32, in term of code block. Each code block is coded independently and the identified overhead is inserted to each data stream.

Figure C.8: Codec for occlusion consideration (a) encoder (b) decoder
Figure C.8 (a) and (b) show the encoder and the decoder in which the occlusion is considered with EBCOT algorithm in stereo image compression, respectively. Fundamentally, the original image with pixel value $p_{ij}$ is applied to the disparity estimation and compensation block for prediction by using information from the reference image, $p'_{ij}$. The output, a residual image, is transformed and quantized, and then the EBCOT encoder is exploited to compress image data. In occlusion coding, the original image is predicted the occluded region and the whole image is biased with value $-2^B - 1$ in order to offset unsigned values to signed values around zero center, where $B$ is the number of bits in each pixel. The blocks with values, $p'_{ij}$, which are marked as the occlusion, $O_{ij}$, are selected for substituting into the same location in the residual picture, $R_{ij}$. Finally, this interleaved image is transformed, quantized and coding via EBCOT and give the output $R_{ij}^O$, while the disparity vectors, $DV$, and indicated occlusion overhead, $O_{ij}$, are coded together via the Huffman coding.

For decoder, the data streams are operated reversely from the encoder. After taking Tier-1 and Tier-2 decoding, dequantization and IDWT, the picture is separated into coded residual blocks, $\tilde{R}_{ij}$ and occlusion blocks, $\tilde{p}_{ij}$, by using the indicated occlusion overhead, $O_{ij}$. The residual blocks are applied to invert disparity estimation and compensation block for reconstructing the original image back, while the occlusion blocks are offset with $2^B - 1$. The reconstruction of the whole image is done by combining these two parts and the final reconstructed image is come out.