Video Transcoding
for Multimedia Communication Networks

by

Safak Dogan

Thesis submitted for the Degree of
Doctor of Philosophy
from the
University of Surrey

Unis

Centre for Communication Systems Research (CCSR)
School of Electronics, Computing and Mathematics
University of Surrey
Guildford, Surrey GU2 7XH, U.K.

October 2001

© Safak Dogan 2001
to my beloved mum, Ferhan;

dad, Necdet;

&

brother, Ediz Dogan
Abstract

Video transcoding is a generic name for a video gateway structure whereby the tandeming process does not involve any high complexity cascaded decoding and re-encoding operations in contrast to the existing conventional solutions. Diverse multimedia communication network characteristics, such as bandwidth limitations and varying congestion conditions, incur quality degradation in video transmissions. The matching of input and output network constraints and characteristics is possible with video transcoding at a centralised unit within the network. Moreover, video transcoding also provides a suitable translation mechanism for different video compression standards achieving a transparent interconnection between diverse network topologies. In addition, video transcoding offers a method for providing robustness against transmission error effects which occur during the transmission of compressed video streams over highly bandwidth-restricted communication media, such as popular mobile-wireless networks. Due to the severe bandwidth restrictions of such networks, the video signals require low bit rate coding which in turn renders video streams highly susceptible to radio channel errors. Therefore, error-resilient operations also need to be provided together with the syntax and transmission rate translation features of video transcoders.

The unique features of video transcoding provide flexible and efficient ways to alleviate the previously addressed three major issues for various requirements. These requirements can be imposed by the diversity of networks on which numerous applications are running or by different standards themselves as well as by a wide range of users. This is strictly related to a universal interoperability issue of heterogeneous characteristics and requirements which demand effective end-to-end solutions. Thus, the aim of this research presented in this thesis is to develop algorithms for the provision of such remedial solutions to the interoperability problem.

In the light of these facts, the research work focuses on the design of various video transcoding algorithms. The objectives of these algorithms are to ease network congestion and/or user bandwidth limitation conditions, support essential standard conversions between different compression schemes and provide necessary error robustness over highly error-prone transmission media, such as mobile radio networks. The ultimate target is to establish a common platform where all the above three aims are successfully satisfied. Extensive computer simulations demonstrate the effectiveness of the proposed and designed systems throughout the course of the research work. These simulations are assessed with the use of objective and subjective performance measures.
Acknowledgements

I would like to express my sincere gratitude to my PhD project supervisors, Dr Abdul H. Sadka and Prof Ahmet M. Kondoz, for their close concern, help and guidance throughout my research. I am very much indebted to both of my supervisors and also Prof Barry G. Evans whose support has turned my PhD dream to reality. Besides, I would also like to thank all my colleagues in the Multimedia Communications Group of CCSR, especially Stewart Worrall, Akin Cellatoglu and ex-fellow Philip Chr. Psiloinis, for their invaluable help at various stages of my research work.

I also wish to acknowledge the Turkish Education Foundation for providing me the opportunity of doing my MSc degree at the University of Surrey which was a major step to the PhD.

Finally, I would like to take this opportunity to express my gratefulness to my dear family for their endless love, support and encouragement they have provided under all kinds of circumstances. Last but not least, very special thanks also go to a number of close friends, Maria Farrugia, Tolga Ors and Sertac Eminsoy, whose friendship will always be highly valued.
# Contents

Abstract ................................................................................................. i
Acknowledgements ................................................................................ ii
Contents ................................................................................................. iii
List of Figures ........................................................................................ vii
List of Tables ........................................................................................ xii

Chapter 1 Introduction 1
  1.1 Background and Objectives ......................................................... 1
  1.2 Quality Assessment by Performance Evaluation............................ 2
  1.3 Original Achievements ............................................................... 4
  1.4 Thesis Outline ........................................................................... 5

Chapter 2 Low Bit Rate Video Communication Systems 8
  2.1 Introduction ............................................................................... 8
  2.2 The Need for Low Bit Rate Video Communications ..................... 9
    2.2.1 User Requirements Perspective ........................................... 11
      2.2.1.1 Coding Efficiency .......................................................... 11
      2.2.1.2 Perceptual Video Quality ................................................. 11
      2.2.1.3 Delay ............................................................................. 12
      2.2.1.4 Complexity ................................................................. 12
      2.2.1.5 Error Resilience ............................................................ 12
      2.2.1.6 Scaleability ................................................................. 13
      2.2.1.7 Further Requirements .................................................... 13
    2.2.2 Networks Perspective ........................................................ 13
  2.3 Low Bit Rate Block-Based Video Coding .................................... 15
    2.3.1 Block-Based DCT coding ...................................................... 16
      2.3.1.1 Motion Estimation/Compensation ...................................... 17
      2.3.1.2 Discrete Cosine Transform (DCT) ...................................... 17
      2.3.1.3 Quantisation ................................................................. 18
      2.3.1.4 Entropy Coding ............................................................ 18
    2.3.2 Low Bit Rate Video Coding Standards ................................. 19
      2.3.2.1 ISO MPEG-4 ................................................................. 19
      2.3.2.2 ITU H.263 ...................................................................... 21
      2.3.2.3 Unrestricted MV Mode (Embedded in MPEG-4,
<table>
<thead>
<tr>
<th>3.4.4 Transcoding with Motion Data Re-Estimation Scheme</th>
<th>61</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.4.5 Transcoding with Motion Data Refinement Scheme</td>
<td>62</td>
</tr>
<tr>
<td>3.4.5.1 The MV Refinement Algorithm</td>
<td>63</td>
</tr>
<tr>
<td>3.4.5.2 The Effects of Refinement Window Size Variation on Transcoding Quality</td>
<td>66</td>
</tr>
<tr>
<td>3.4.5.3 Tuning of Video Transcoding Quality by Self-Adjusting Search Window Approach</td>
<td>68</td>
</tr>
<tr>
<td>3.5 Frame Rate Reduction</td>
<td>71</td>
</tr>
<tr>
<td>3.6 Resolution Reduction</td>
<td>72</td>
</tr>
<tr>
<td>3.7 Multi-Transcoder Design for Multimedia Traffic Planning</td>
<td>75</td>
</tr>
<tr>
<td>3.7.1 The System Architecture for Traffic Planning Purposes</td>
<td>77</td>
</tr>
<tr>
<td>3.8 Computer Simulations and Analysis of the Results</td>
<td>79</td>
</tr>
<tr>
<td>3.8.1 Bit Rate Reduction</td>
<td>79</td>
</tr>
<tr>
<td>3.8.1.1 Experiments and Results</td>
<td>79</td>
</tr>
<tr>
<td>3.8.1.2 Analysis of the Results</td>
<td>85</td>
</tr>
<tr>
<td>3.8.2 Frame Rate Reduction</td>
<td>86</td>
</tr>
<tr>
<td>3.8.2.1 Experiments and Results</td>
<td>86</td>
</tr>
<tr>
<td>3.8.2.2 Analysis of the Results</td>
<td>87</td>
</tr>
<tr>
<td>3.8.3 Multimedia Traffic Planning</td>
<td>88</td>
</tr>
<tr>
<td>3.8.3.1 Experiments and Results</td>
<td>88</td>
</tr>
<tr>
<td>3.8.3.2 Analysis of the Results</td>
<td>90</td>
</tr>
<tr>
<td>3.9 Concluding Remarks</td>
<td>91</td>
</tr>
</tbody>
</table>

Chapter 4 Heterogeneous Video Transcoding

| 4.1 Introduction                                      | 94 |
| 4.2 Heterogeneous Video Transcoding                  | 95 |
| 4.3 MPEG-4/H.263 Video Transcoding                   | 96 |
| 4.3.1 The Importance of Using Two Different Standards | 96 |
| 4.3.2 Tandem in Asymmetric Networks Scenarios        | 97 |
| 4.4 Types of Heterogeneous Tandem of MPEG-4/H.263     | 98 |
| 4.4.1 Conventional Cascaded Fully Decoding/Re-Encoding | 98 |
| 4.4.2 Bi-Directional Heterogeneous MPEG-4/H.263 Video Transcoding | 99 |
| 4.4.2.1 Differences in Video Frame Headers           | 101 |
| 4.4.2.2 Differences in MV Tables                     | 103 |
| 4.4.2.3 Differences in DCT Coefficient Tables        | 103 |
| 4.4.2.4 Differences in Intra DC Quantisation Methods | 103 |
| 4.4.2.5 Differences in Motion Compensation with Rounding | 103 |
Contents

4.4.2.6 Bi-Directional Heterogeneous MPEG-4/H.263 Video Transcoder Algorithm ................................................................. 105

4.5 Computer Simulations and Analysis of the Results ............................................................................................................. 107

4.5.1 Experiments and Results ...................................................................................................................................................... 107

4.5.2 Analysis of the Results ...................................................................................................................................................... 111

4.6 Concluding Remarks ............................................................................................................................................................. 112

Chapter 5 Error-Resilient Video Transcoding .......................................................................................................................... 114

5.1 Introduction .................................................................................................................................................................................. 114

5.2 Error-Resilient Video Transcoding Requirement Scenarios .................................................................................................. 117

5.2.1 General Packet Radio Service (GPRS) Networks .................................................................................................................. 119

5.3 Error-Resilient Video Transcoder Architecture .................................................................................................................... 122

5.4 Computer Simulations and Analysis of the Results .............................................................................................................. 125

5.4.1 Transcoding with AIR ........................................................................................................................................................... 125

5.4.1.1 Experiments and Results for Random Error Effects .................................................................................................... 125

5.4.1.2 Analysis of the Results .................................................................................................................................................... 127

5.4.1.3 Experiments and Results for Burst Error Effects .......................................................................................................... 129

5.4.1.4 Analysis of the Results .................................................................................................................................................... 135

5.4.2 Transcoding with FCS ........................................................................................................................................................... 137

5.4.2.1 Experiments and Results with Heterogeneous Transcoding ......................................................................................... 137

5.4.2.2 Analysis of the Results .................................................................................................................................................... 141

5.4.2.3 Experiments and Results with Homogeneous Transcoding .......................................................................................... 142

5.4.2.4 Analysis of the Results .................................................................................................................................................... 150

5.5 Concluding Remarks ............................................................................................................................................................. 152

Chapter 6 Conclusion ......................................................................................................................................................................... 155

6.1 Preamble ..................................................................................................................................................................................... 155

6.2 Concluding Overview ............................................................................................................................................................... 156

6.3 Thoughts on Future Work ......................................................................................................................................................... 161

Appendix A List of Publications ..................................................................................................................................................... 164

Appendix B List of Abbreviations .................................................................................................................................................. 166

Bibliography .................................................................................................................................................................................. 169
List of Figures

Chapter 2

Figure 2.1 A block-based video encoder block diagram ........................................................... 15
Figure 2.2 Motion prediction in inter frame coding ................................................................. 17
Figure 2.3 Classification of bit rates and functionalities supported by MPEG-4 ...................... 20
Figure 2.4 Candidate predictors for MV1, MV2 and MV3 for advanced prediction mode .......... 23
Figure 2.5 Error detection in VLC video bitstreams ................................................................. 27
Figure 2.6 The operation of the FCS algorithm for a 2-frame round-trip delay scenario .......... 30
Figure 2.7 AIR method for a QCIF video frame ....................................................................... 31
Figure 2.8 The original first frames of the video test sequences used in the thesis: “Akiyo_with_Crowd”, “Carphone”, “Claire”, “Container”, “Foreman”, “Miss_America”, “Mother & Daughter”, “Salesman”, “Sign_Irene” and “Suzie”, respectively ................................................................. 32
Figure 2.9 The objective results of H.263 direct enc/dec a- “Suzie” and b- “Foreman” with and without options ........................................................................................... 34
Figure 2.10 The comparative objective results of H.263 and MPEG-4 direct enc/dec a- “Suzie” and b- “Foreman” (with options) ................................................................................ 34
Figure 2.11 The subjective results of the 150th frames of “Suzie”, a- original; b- H.263 enc/dec (without options); c- H.263 enc/dec (with options); d- MPEG-4 enc/dec (with options) ................................................................................ 35
Figure 2.12 The subjective results of the 200th frames of “Foreman”. Top row- H.263 w/out options; middle row- H.263 with options; bottom row- MPEG-4 with options. First column- QP=4; second column- QP=6; third column- QP=11; fourth column- QP=20 ....................................................... 35
Figure 2.13 The objective results (the lost frame numbers: 6, 22, 81, 108, 127 and 188) ..................................................................................................................... 39
Figure 2.14 The subjective results of the 200th frames of “Container”, “Foreman” and “Claire” ............................................................................................................... 40
Figure 2.15 The subjective results of a- “Suzie” and b- “Miss_America” (BER = 1e-03) ....................... 42
Figure 2.16 The subjective results of the 150th frames of “Suzie” over a BER = 1e-03 random error channel, a- error-free; b- non-resilient error-prone and c- error-resilient with AIR ....................................................... 43
Figure 2.17 The subjective results of the 150th frames of “Miss_America” over a BER = 1e-03 random error channel, a- error-free; b- non-resilient error-prone and c- error-resilient with AIR ............... 43
Chapter 3

Figure 3.1 A multimedia networking scenario using a video transcoder at the video gateway ........................................................................... 48
Figure 3.2 Video transcoding ................................................................................................................................. 49
Figure 3.3 Homogeneous video transcoding ........................................................................................................... 51
Figure 3.4 Cascaded fully decoding/re-encoding scheme versus transcoding ......................................................... 54
Figure 3.5 Transcoding with re-quantisation scheme .............................................................................................. 55
Figure 3.6 A basic drift-free homogeneous video transcoder block diagram ............................................................. 58
Figure 3.7 Simplified drift-free video transcoder block diagram ............................................................................. 59
Figure 3.8 Transcoding with motion data re-use scheme ......................................................................................... 61
Figure 3.9 Transcoding with motion data re-estimation scheme ............................................................................. 62
Figure 3.10 Transcoding with motion data refinement scheme ................................................................................. 63
Figure 3.11 A complete video transcoding system ................................................................................................. 64
Figure 3.12 The methodology of MV refinement .................................................................................................... 66
Figure 3.13 Average PSNR and bit rate variations of the transoded “Foreman” sequence, originally encoded at an average of 116 kbit/s and 25 fr/s, versus various refinement window sizes ........................................................................... 67
Figure 3.14 Average PSNR and bit rate variations of the transoded “Suzie” sequence, originally encoded at an average of 64 kbit/s and 25 fr/s, versus various refinement window sizes ........................................................................... 67
Figure 3.15 Block diagram of a video transcoder with refining window adjuster ............................................................. 69
Figure 3.16 MV handling during frame rate conversion ............................................................................................ 71
Figure 3.17 Down-sampling by a factor two in each dimension, a- candidate MV selection problem; b- candidate MB type (mode) selection problem ............................................................................. 73
Figure 3.18 A typical resolution transcoding scenario ............................................................................................... 74
Figure 3.19 MPEG-4 multiparty video-telephony scenario, using a video transcoder at the MCU for multimedia traffic planning .......................................................................................................................... 76
Figure 3.20 System architecture; the video transcoder bank (superscripts i and o representing the inputs and outputs, respectively and subscripts representing the network types) ............................................................................. 77
Figure 3.21 Objective test results for the open-loop video transcoding of the “Suzie” sequence, a- PSNR and b- bit rate variations .......................................................................................................................... 80
Figure 3.22 Objective test results for the closed-loop video transcoding of the “Suzie” sequence, a- PSNR and b- bit rate variations .......................................................................................................................... 80
Figure 3.23 Objective test results for the homogeneous video transcoding of the various video sequences, a- “Suzie” (QP=8→14) (I- PSNR, 2- bit rate); b- “Foreman” (QP=8→14) (I- PSNR, 2- bit rate); c- “Sign_Irene” (QP=7→17) (I- PSNR, 2- bit rate); d- “Carphone” (QP=8→14); e- “Akiyo_with_Crowd” (QP=5→12); f- “Foreman” (QP=20→30); g- “Suzie” (QP=16→23) ........................................................................................... 82
Figure 3.24 Subjective results of the last frames of the “Suzie”, “Foreman”, “Carphone”, “Sign_Irene” and “Akiyo_with_Crowd” sequences, a- direct enc/dec (at higher bit rate); b- direct enc/dec (at lower bit rate); c- re-quantisation scheme (1- motion data is ignored, 2- motion data is not ignored); d- cascaded fully dec/re-enc scheme; e- MV re-use scheme; f- MV refinement scheme ........................................ 83

Figure 3.25 PSNRs of the layered transcoder bank outputs for “Foreman” ...................................... 88

Figure 3.26 Objective results of the “Foreman” sequence for the adaptive multimedia traffic planning, a- PSNR and b- bit rate variations ................................................ 89

Figure 3.27 Objective results of the “Suzie” sequence for the adaptive multimedia traffic planning, a- PSNR and b- bit rate variations ................................................ 89

Chapter 4

Figure 4.1 Simplified block diagram of a tandem of the two video communication standards .................. 95

Figure 4.2 Heterogeneous multimedia networks scenarios................................................................ 98

Figure 4.3 The conventional way (cascaded fully decoding/re-encoding) versus transcoding................. 99

Figure 4.4 Heterogeneous MPEG-4/H.263 video transcoding essentials ............................................ 101

Figure 4.5 Bitstream syntax of the MPEG-4 standard ........................................................................ 102

Figure 4.6 Bitstream syntax of the H.263 standard ........................................................................... 102

Figure 4.7 Bi-linear interpolation scheme for half sample search for both standards ......................... 104

Figure 4.8 Bitstream syntax mapping between the two standards ..................................................... 105

Figure 4.9 PSNR variations with and without transcoding for the 150-frame “Suzie” sequence at 56 kbit/s on average with a frame rate of 25 fr/s ........................................ 108

Figure 4.10 Subjective results of the 150th frame of the “Suzie” (56 kbit/s on av.) sequence. Top row- MPEG-4 to H.263 path; bottom row- H.263 to MPEG-4 path. First column- direct encoding/decoding; second column- transcoding; third column- cascaded decoding/re-encoding .................................. 108

Figure 4.11 PSNR variations with and without transcoding for the 200-frame “Carphone” sequence at 82 kbit/s on average with a frame rate of 25 fr/s ........ 109

Figure 4.12 Subjective results of the 200th frame of the “Carphone” (82 kbit/s on av.) sequence. Top row- MPEG-4 to H.263 path; bottom row- H.263 to MPEG-4 path. First column- direct encoding/decoding; second column- transcoding; third column- cascaded decoding/re-encoding .................................. 109

Figure 4.13 PSNR variations with and without transcoding for the 200-frame “Foreman” sequence at 100 kbit/s on average with a frame rate of 25 fr/s ........ 110

Figure 4.14 Subjective results of the 200th frame of the “Foreman” (100 kbit/s on av.) sequence. Top row- MPEG-4 to H.263 path; bottom row- H.263 to MPEG-4 path. First column- direct encoding/decoding; second column- transcoding; third column- cascaded decoding/re-encoding .................................. 110
Chapter 5

Figure 5.1 Error-resilient video transcoding scenario............................................................... 115
Figure 5.2 Error-resilient video transcoding with error concealment scenario.......................... 116
Figure 5.3 A heterogeneous multimedia networking scenario using a video transcoder at the video gateway................................................................................................118
Figure 5.4 A GPRS networking scenario with an error-resilient video gateway .......................119
Figure 5.5 Usage of shared pool of GPRS resources............................................................... 121
Figure 5.6 The error-resilient video transcoding architecture....................................................123
Figure 5.7 Objective results of the 150-frame “Suzie” sequence at near 39 kbit/s on average, a- PSNR versus BER and b- PSNR versus frame number for a BER = 1e-03. (The results shown are the averages of 10 simulations run with different seeds)......................................................126
Figure 5.8 Objective results of the 200-frame “Mother & Daughter” sequence at near 27 kbit/s on average, a- PSNR versus BER and b- PSNR versus frame number for a BER = 1e-03. (The results shown are the averages of 10 simulations run with different seeds) ..........................................................126
Figure 5.9 Subjective results of the 150th frames of “Suzie” for a particular seed at BER = 1e-03, a- error-free; b- non-resilient error-prone and c- error-resilient sequences ..........................................................127
Figure 5.10 Objective results of the 200-frame “Mother & Daughter” sequence at near 27 kbit/s on average for CS1, CS2 and CS3, a- PSNR versus C/I and b- PSNR versus frame number for a C/I = 12 dB. (The results shown are the averages of 10 simulations run with different seeds) ........................................................................132
Figure 5.11 Subjective results of the 200th frames of “Mother & Daughter” for a particular seed at C/I = 12 dB, a- error-free direct enc/dec at 70.553 kbit/s; b- CS1, non-resilient error-prone transcoded down to 25.818 kbit/s; c- CS2, non-resilient error-prone transcoded down to 25.818 kbit/s; d- CS3, non-resilient error-prone transcoded down to 25.818 kbit/s; e- error-free transcoded down to 25.818 kbit/s; f- CS1, error-resilient transcoded down to 27.000 kbit/s; g- CS2, error-resilient transcoded down to 27.000 kbit/s; h- CS3, error-resilient transcoded down to 27.000 kbit/s sequences..........................133
Figure 5.12 Objective results of the 200-frame “Foreman” sequence at near 47 kbit/s on average for CS1, CS2 and CS3, a- PSNR versus C/I and b- PSNR versus frame number for a C/I = 12 dB. (The results shown are the averages of 10 simulations run with different seeds) ........................................................................134
Figure 5.13 Subjective results of the 200th frames of “Foreman” for a particular seed at C/I = 12 dB, a- error-free direct enc/dec at 87.403 kbit/s; b- CS1, non-resilient error-prone transcoded down to 46.835 kbit/s; c- CS2, non-resilient error-prone transcoded down to 46.835 kbit/s; d- CS3, non-resilient error-prone transcoded down to 46.835 kbit/s; e- error-free transcoded down to 46.835 kbit/s; f- CS1, error-resilient transcoded down to 46.986 kbit/s; g- CS2, error-resilient transcoded down to 46.986 kbit/s; h- CS3, error-resilient transcoded down to 46.986 kbit/s sequences........................................135
Figure 5.14 Error-resilient heterogeneous video gateway application scenarios ......................138
Figure 5.15 Objective test results of “Suzie” (the lost frame numbers: 9, 17, 52,
List of Figures

77, 100, 133, 142), a- PSNR and b- bit rate variations .............................................. 139

Figure 5.16 Objective test results of "Claire" (the lost frame numbers: 13, 34, 69, 111, also 112 for the 2-consecutive frame loss case, 145, 163, 178), a- PSNR and b- bit rate variations ........................................................................................................ 140

Figure 5.17 Subjective results of the 56th frames of "Suzie", a- error-free; b- non-resilient error-prone (5% frame loss) and c- 2-frame delay error-resilient (5% frame loss) sequences .................................................................................. 140

Figure 5.18 Subjective results of the 116th frames of "Claire", a- error-free; b- non-resilient error-prone (3.5% frame loss); c- non-resilient error-prone with 2-consecutive frame loss and d- 2-frame delay error-resilient (3.5% frame loss) sequences ............................................................................... 140

Figure 5.19 Round-trip delay of the feedback signal over GPRS ........................................ 143

Figure 5.20 Objective results of "Suzie", average PSNR variations against a- feedback signalling round-trip delay and b- number of transcoded frames .................................................................................. 143

Figure 5.21 Objective results of "Salesman", average PSNR variations against a- feedback signalling round-trip delay and b- number of transcoded frames .................................................................................. 144

Figure 5.22 Objective results of "Foreman", average PSNR variations against a- feedback signalling round-trip delay and b- number of transcoded frames .................................................................................. 144

Figure 5.23 Subjective results of the 150th frames of "Suzie" and "Salesman" and 200th frames of "Foreman", respectively, a- error-free; b- 2-frame delay; c- 6-frame delay; d- 10-frame delay resilient; e- error-prone; f- 4-frame delay; g- 8-frame delay; h- 12-frame delay resilient sequences .................................................................................. 146

Figure 5.24 Objective results for the combination of AIR and FCS over a C/I = 12 dB CS2 GPRS channel model, a- "Suzie"; b- "Salesman" and e- "Foreman" sequences requiring 3, 4 and 5 timeslots, respectively .................................................................................. 147

Figure 5.25 Subjective results of the 67th (a-f) and 150th (g-l) frames of "Suzie", 150th (m-p) frames of "Salesman" and 200th frames of "Foreman", a/f/h/l/- error-free; b/i/o/s- error-prone; c/j/- FCS only resilient; d/k/- AIR only resilient; e/l/p/- FCS and AIR combined resilient; f/g/m/q/- direct enc/dec at higher rate sequences over a C/I = 12 dB CS2 GPRS channel model requiring 3, 4 and 5 timeslots, respectively .................................................................................. 148

Chapter 6

Figure 6.1 A prospective networking scenario with the use of video, user and multimedia gateways .............................................................................................................. 163
List of Tables

Chapter 2

Table 2.1 The summary of the objective results of “Suzie” and “Foreman” ......................... 36
Table 2.2 Average PSNR and bit rate values of the three test sequences ......................... 40
Table 2.3 The detailed PSNR and bit rate variations for both test sequences ..................... 43

Chapter 3

Table 3.1 PSNRs of the various test sequences for different bit rate conversions ............. 69
Table 3.2 Transcoded bit rates, “Suzie” from 64 kbit/s and “Foreman” from 116 kbit/s on average .................................................. 70
Table 3.3 Percentage of number of refined MBs to number of overall MBs .................... 70
Table 3.4 The effects of the I-frame transcoding and the motion data care during open-loop transcoding ........................................................................ 80
Table 3.5 Bit rate transcoding results for various video test sequences ........................ 84
Table 3.6 Frame rate transcoding results for various video test sequences .................. 87
Table 3.7 Average bit rate and PSNR values for Figure 3.25 ............................................ 89
Table 3.8 Summary of the advantages/disadvantages of the five bit rate transcoding schemes .................................................................................. 92

Chapter 4

Table 4.1 Inter-standard differences between H.263 and MPEG-4 .............................. 100
Table 4.2 The distinctive features of the two tandem methods for a quick comparison ..... 113

Chapter 5

Table 5.1 GPRS channel coding schemes ................................................................. 120
Table 5.2 Average PSNR and bit rate values against different BERs (the first value in each cell before the semi-colon represents the PSNR in dB and the second value shows the bit rate in kbit/s) .................................................. 127
Table 5.3 Timeslotting capability and the raw user data rates employed for the experimentation of the different GPRS channel protection schemes ......... 130
List of Tables

Table 5.4 Average PSNR and bit rate values against different C/I ratios .................................. 132
Table 5.5 Experimentally obtained BERs against C/I for "Mother & Daughter" for different channel protection schemes over GPRS ......................................................... 132
Table 5.6 Average PSNR and bit rate values against different C/I ratios .................................. 134
Table 5.7 Experimentally obtained BERs against C/I for "Foreman" for different channel protection schemes over GPRS ................................................................. 134
Table 5.8 The video quality improvements by the error-resilient transcoding over the non-resilient scheme .......................................................................................... 136
Table 5.9 Average bit rate and PSNR values for the two test sequences .................................. 141
Table 5.10 Average bit rate and PSNR values for "Suzie", "Salesman" and "Foreman" ...................... 146
Table 5.11 Average bit rate, PSNR and BER values for "Suzie", "Salesman" and "Foreman" over a C/I = 12 dB CS2 GPRS channel model requiring 3, 4 and 5 timeslots, respectively ................................................................. 149

Chapter 6

Table 6.1 The summary of concluding remarks of thesis chapters for a quick reference .... 161
Chapter 1

1 Introduction

1.1 Background and Objectives

Video transmission has gained great deal of pace lately due to its importance to the entertainment world. This importance forced the video technology to advance with head-spinning speed particularly in the last decade of the 20th century. As a result of the latest technological advances, it has also become possible to efficiently compress the bandwidth-hungry video signals to transport even over bandwidth-limited mobile-wireless systems. Separate media communications have therefore evolved into multimedia communications with the combination of individual video, speech, audio and data communications. Today, multimedia is one of the best and the most significant revenue expectations of many telecommunication companies and hence, it is also one of the leading investments for the future. Thus, the 21st century will experience many more advances in video communications.

It is indeed not a very big surprise that video communications have received this much attention as they have potential for a great many varieties of applications. Besides the initiative role in the entertainment industry, video communications have been gaining considerable interest for numerous other practices, such as video-telephony/conferencing, distant education, remote health monitoring, security surveillance, remote video database access as well as interactive multimedia games for pure entertainment.

Meanwhile, the latest perceptions on the importance and popularity of mobile communications initiated a rapid deployment of mobile-multimedia services. However, these services have penetrated into human life with their related problems, such as significant requirements for low bit rate compression, error protection or robustness, interconnectivity and compatibility to existing standards. Thus, the search for urgent remedies to these problems, which were once associated to the fixed services, such as video-on-demand (VoD) and broadcast applications, has swiftly been commenced. The growing demand towards the access to the multimedia data on the move forced these remedies to be taken efficiently apart from the urgency factor. One of the most significant problems has appeared to be the interoperability of numerous applications with their vast number of diverse requirements. Therefore, this very state-of-the-art concept has also led this research
work carried out here and guided this particular thesis to discuss and investigate the interoperability issues for low bit rate video communications.

Since the standards and hence, the applications relying on these standards cannot be made future-proof, the gateways connecting different standards should be designed to withstand prospective technologies. For this reason, interconnection of various user interfaces within different multimedia networks has also become notable as well as video compression and transmission technologies. Thus, the main objective of this thesis is to develop efficient methods which link numerous video end-user needs whilst also taking their application driven constraints and the varying network conditions into consideration. The techniques, namely video transcoding algorithms, are defined and elaborated in this thesis. Furthermore, improving the transcoded video quality in error-prone environments with error resilience schemes is also aimed. In this way, it is believed that the design of an error-robust interoperability system will be able to provide an ultimate interconnection mechanism for the future third generation (3G) mobile-multimedia networks. In summary, the objectives of the research work is to establish the basics for a generic video transcoding algorithm which proposes solutions to alleviate the following interoperability problems:

- Diverse congestion or bandwidth limitation characteristics.
- Various syntax requirements imposed by different video compression schemes.
- Varying quality of service (QoS) properties due to channel errors.

The rest of the chapter is organised as follows: The second section discusses how the performance evaluations were carried out in the thesis. The third section presents the original achievements of this research whilst the fourth section focuses on the outline of the thesis.

### 1.2 Quality Assessment by Performance Evaluation

The performance evaluations of the results obtained from the developed algorithms are accomplished with the use of two methods throughout the thesis: objective and subjective quality assessments. These two techniques are utilised due to their wide use in video coding and communication research environments. Thus, a consistent comparison with the results demonstrated by other researchers is aimed.

The first method employed for the quality assessment is an objective measure which is computed by using some mathematical criteria. Conventionally, two types of criteria have been used, such as peak-to-peak signal-to-noise ratio (PSNR) and mean-square-error (MSE). Both methods have common features for their objective image quality measures as they rely on individual pixel values
of the original and reconstructed video frames. Thus, this kind of an assessment has got nothing to
do with the quality perception of the human visual system (HVS). Between the two, PSNR is
commonly adopted for objective measurement and has widely been used by the research
community although MSE is still occasionally used. The mathematical representations of PSNR
and MSE are given in Equations (1.1) and (1.2) as follows:

\[
PSNR = 10 \log_{10} \frac{255^2}{\frac{1}{M \times N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} [x(i, j) - \hat{x}(i, j)]^2}
\]  

(1.1)

\[
MSE = \frac{1}{M \times N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} [x(i, j) - \hat{x}(i, j)]^2
\]  

(1.2)

where the maximum value of a pixel with an 8-bit representation can be 255. From these two
equations it can be observed that MSE is an integral part of the PSNR calculation which computes
the pixel impairments within an entire image. Here, \(M\) and \(N\) stand for the sizes of the image
whereas \(x\) and \(\hat{x}\) are the original and reconstructed pixels, respectively. \(M\) and \(N\) correspond to
176\times144 pixels in the case of a quarter common intermediate format (QCIF) image quality
evaluation.

The second method of video quality assessment comprises the subjective image comparison tests.
Practically, since the HVS does not quite work like a computer which assesses the quality pixel-
by-pixel, subjective measurements are also needed. This measurement is generally made on the
whole of an image, not on every single pixel of the image. Consequently, the overall picture
quality is assessed in this way. It is quite significant that the objective results are also supported
by this kind of subjective results as HVS is generally more sensitive to the area of interest in a
video image rather than the still or occasionally changing background images. Moreover, a second
video object in the entire sequence may even not be the focus of the concern to an individual
observer. For instance, in a case that an error resilience scheme is adopted to improve the quality
only for the regions of interest, the subjective tests gain importance compared to the low quality
objective results. This is due to the fact that the objective results are computed using the PSNR
method which does not distinguish particular regions of interest within a picture. Thus, during the
research work, objective quality results are accompanied by the subjective quality assessments. In
the latter method, a reference sequence and an output reconstructed sequence from a proposed
algorithm are displayed side by side for evaluation. Although these tests are carried out for the
entire sequences, only the last frames of each of the assessed video clips can be demonstrated in
the thesis. The presentation of the results comprises PSNR objective results and the pair
comparison subjective results.
Lastly, for a fair performance evaluation of a video coding algorithm, the bit rate is also mentioned for each of the quality assessment tests in the thesis. The output bit rate of a video coder is expressed in bits per second [bit/s]. In addition, since the bit rate is directly proportional to the number of image pixels and number of frames per second [fr/s], both image sizes and the frame rates for the tested video clips are also indicated during the performance evaluation processes.

1.3 Original Achievements

The work presented in this thesis is primarily based on the current International Telecommunication Union (ITU) H.263 [H263] and International Standards Organisation (ISO) Motion Picture Experts Group (MPEG) MPEG-4 Visual [MPEG4] standards for low bit rate video coding. The utilised video encoding/decoding sets are the Test Model (TMN) TMN-1.6 H.263 and Version V-1 of Mobile Multimedia Systems (MoMuSys) Verification Model (VM) VM-8.0 MPEG-4 visual coding algorithms. The research work undertaken aims at the development of generic video transcoding algorithms for interoperability issues of the diverse heterogeneous multimedia networks with error resilience capabilities. This work has generated three journal and three conference papers with a pending contribution to a book chapter and a recent submission to a journal, which is still in the reviewing process. A more detailed list of publications is given in Appendix A of the thesis. Furthermore, the work which is believed to be original and contributory to achieve the objectives can be summarised as follows:

* The design of a novel adaptive multimedia traffic management algorithm comprising a bank of video transcoders operating in parallel to each other which eases the varying network congestion conditions.

* The design of a novel efficient MPEG-4/H.263 video transcoding algorithm which fully interconnects the two low bit rate video coding standards in the compressed domain bi-directionally. This technique avoids the conventional cascaded fully decoding and re-encoding operations.

1.4 Thesis Outline

Chapter 1 has comprised an introduction to the thesis presenting the background and basis of the undertaken research work. In this chapter, the objectives of the work have been defined. To meet the objectives, quality assessment techniques, which are used throughout the thesis, have been explained. Moreover, this chapter has also presented the original contributions that are believed to be achieved during the course of the research. Finally, the chapter has ended with this section which outlines the thesis structure comprising brief introductory paragraphs dedicated to each of the chapters.

In Chapter 2, an introduction to the low bit rate video communication systems is considered. The reasons behind the necessity of low bit rate video communications are examined with the presentation of the user and network perspectives. Furthermore, low bit rate video coding algorithms are summarised. Two low bit rate video coding standards, namely H.263 and MPEG-4, are also introduced. Then, error effects on the compressed video has been considered whilst also discussing the resilience methods as to mitigate the destructive impacts of errors in video transmissions. In this particular part, two error resilience techniques are discussed in detail which are further exploited for error-resilient video operations in the thesis. Moreover, the source material used for the performance evaluations of the proposed algorithms throughout the thesis is introduced. Lastly, computer simulations, which are directly related to pure low bit rate video coding techniques and error resilience methods, are also demonstrated and their results are analysed.

Chapter 3 presents a general introduction on the video transcoding technologies which provide necessary background knowledge for the succeeding discussions in the forthcoming contents of the thesis. Here, the video transcoding requirements due to the heterogeneity of the multimedia networks and various categories of transcoding algorithms are summarised. This is followed by a detailed investigation of the first category of video transcoding methods. Subsequent to a brief overview, the three different types of this kind of a transcoding achievement, namely bit rate reduction, frame rate reduction and resolution reduction schemes, are described. Video transmission (bit and frame) rate reduction algorithms are discussed in a very much detailed way whilst only a brief overview is given for the resolution reduction methods as these methods are beyond the scope of this research work and the thesis. In this chapter, a rate regulating MPEG-4 standard-compliant video transcoding algorithm is gradually developed from a primitive conventional cascaded decoder/re-encoder pair. Thus, the resulting transcoding method is discussed to be a lower complexity and better quality achievement. During the gradual build-up of the ultimate video transcoder, every step-stone has been presented and the resulting algorithms are compared against both the conventional and the ultimate methods of transcoding. Moreover, with
the use of the final transcoding scheme, a video transcoder bank is designed for multimedia video traffic planning purposes. The adaptive mechanism of this kind of a transcoder bank is presented to achieve better qualities even at congestion conditions of multimedia networks. These observations are supported by the computer simulation results demonstrated in the ending part of the chapter.

Chapter 4 focuses on the second category of video transcoding technologies. Thus, in this chapter, a video transcoding algorithm is developed to perform the necessary syntax translations and hence, the required interoperability of two different low bit rate video coding standards, namely H.263 and MPEG-4. Similar to the previous chapter layout, the ultimate transcoding operation is also compared to the conventional way of interconnection: cascaded decoding with one standard and re-encoding with the other. Several tests on the performance show that the developed architecture presents a lower complexity and better quality transcoding operation with a minimum processing delay. These observations are obtained in the computer simulations section of the chapter. Prior to this section, the necessary algorithmic detail is given for the design of a syntax converting video transcoder.

In Chapter 5, the third category of video transcoding is investigated and elaborated. Thus, this chapter gives a detailed insight into an error-resilient video transcoding operation method and the requirements for such an operational mode. This mode of operation is provided with the use of the two error resilience techniques already presented in Chapter 2. However, these methods were directly applied to source coding algorithms in that particular chapter whereas they are an integral part of the video transcoding algorithm here. The error-resilient video transcoding architecture is discussed in detail. Several networking scenarios are set to test the resilient video transcoder. Transcoding performance is tested in random and burst error-prone environments, over transmission channels where video packet loss ratios are high and most significantly over 3G mobile-access networks. In the computer simulations sections, the output of the rate- and resilience-adaptive transcoder is demonstrated to give superior video qualities in severe error-prone conditions. The efficiency of the developed algorithm is also proved by several objective and subjective results. Moreover, this chapter also presents an overview on the 3G mobile-access networks as they are the main concern of the robust video transcoding experimentation. Lastly, the error resilience power of the developed video transcoder is enhanced with the combination of the two separately discussed resilience algorithms.

In addition to the previous discussions, each chapter ends with a concluding section where related sets of comments and concluding remarks are deduced. And finally, in the last chapter, the research work is summarised, observations are further discussed and possible directions for future
research in the area are proposed. At the end, an appendix shows a list of publications that were accomplished in the course of the research work.
Chapter 2

2 Low Bit Rate Video Communication Systems

2.1 Introduction

The rapidly increasing interest in multimedia communications has been prominently driven by the success of efficient video compression standards. Since video has always been known as a very much bandwidth demanding application in its raw data format, the challenge has always been to find a suitable method of compressing it prior to transmission. The amount of compression is an important design parameter which depends both on the bandwidth factor of the communication medium and the minimum video quality required. Lately, these two factors particularly forced the standardisation organisations and the industrial manufacturers to produce higher compression ratios due to the wide deployment of mobile-wireless communications. Nowadays, there is a need rather than a luxurious desire to access the multimedia data on the move. Therefore, the multimedia communications comprising video transmissions have rapidly taken off.

The reason why high compression ratios on video information are needed is that wireless transmission media are extremely bandwidth-limited. Moreover, the access demand to the scarce resources is enormous from a wide range of users. Thus, low bit rate video compression mechanisms help remove the redundancy from the raw video data, which indeed does not need to be transmitted as it can easily be re-generated at the receiving end. However, the removal of redundant information from the video signal renders the video transmission extremely error-sensitive over wireless channels. Thus, for a reliable transmission in an error-prone communication medium, addition of robustness to a video stream is required.

Therefore, this chapter presents two methods of error-resilient low bit rate video communications. However, before the error resilience discussions, low bit rate video communications are further elaborated. Chapter 2 is organised as follows: The second section focuses on the requirement of the low bit rate video communications in detail. The third section presents an insight into the low bit rate block-based video coding; methods and standards. The fourth section reviews the error effects and error resilience factors in video communications along with two particular algorithms.
The fifth section introduces the source material used during the research work for various video performance tests. The sixth section demonstrates the experiments and computer simulation results associated with the tests of pure low bit rate video coding and error-resilient operations. Finally, the seventh section concludes the chapter.

2.2 The Need for Low Bit Rate Video Communications

Before finding an answer to the question “why low bit rate video communications”, it is important to have a brief insight to what other requirements and remedies exist in the video communication world. The integration of moving video as an integral part of multimedia environment is technologically one of the most demanding tasks. Unlike data signals, video information contains redundancies in both time and space. Thus, the digital representation of visual information in its natural form generates a huge amount of data. On the other hand, bandwidth is a major bottleneck for multimedia communications, particularly for the video transmissions whereby raw video data is always hungry for the bandwidth. For instance, a 176x144 QCIF 4:2:0 colour video sequence requires over 1 Mbit/s. Applications running on wireless networks in particular suffer more from the bandwidth requirements and the channel constraints. Therefore, digital video compression is one of the key issues in video communications, enabling efficient interchange and distribution of visual information. Video compression techniques are exploited to remove the redundancies from the video allowing reduced bit rates and feasible transmission solutions. Thus, video compression has to be applied before the video stream is conveyed through to the bandwidth-limited communication channels.

The amount of necessary compression is also an application-dependent decision. This decision aims at the successful provision of the video services to user expectations whilst also complying with the requirements of the selected communication medium. Therefore, the choice of a compression degree is greatly affected by the targeted multimedia application type and its transport over a transmission link. Thus, a number of video coding algorithms have been developed so far to soothe various requirements whilst giving rise to several supported compression depths. Amongst all, the most successful and the most well-known ones are H.261, H.263 and its annex extensions, such as H.263+ and H.263++, of ITU, and MPEG-1, MPEG-2 and MPEG-4 of ISO. Both H.261 [H261] and H.263 [H2631] are video-conferencing oriented low and very low bit rate video coders, respectively. The two extensions of H.263, namely H.263+ [H2632] and H.263++ [H2633], are also in line with their predecessor, H.263, but with extended capabilities. MPEG-1 [MPEG1] is the standard for coding of moving pictures and associated audio for digital storage media at up to 1.5 Mbit/s whilst MPEG-2 [MPEG2] is the standard for
generic coding of moving pictures and associated audio at much higher bit rates. Finally, MPEG-4 [MPEG4] is the ISO's recently released very low to high bit rate, object-oriented video coding standard with superior functionalities compared with its predecessors: MPEG-1 and MPEG-2 [Reade].

It is certain that there exists a video coding and compression standard dedicated for different criteria. Storage requirements are alleviated by the MPEG-1 standard whereas the newly established MPEG-7 [MPEG7] is more suitable for retrieval from the storage medium. On the other hand, MPEG-2 is aimed for broadcast operations whilst H.261, H.263 and MPEG-4 are widely used for low rate transmissions. Finally, for reliable communications over quality destructive media, H.263+, H.263++ or MPEG-4 are available. Since there are these many widely deployed and eminently acknowledged successful video coding and compression standards in our lives, now the question evolves from “why low bit rate video communications” to “where and when low bit rate communications”.

The rapid proliferation of digital video results from the great technological progress that has taken place during the past decade. The current state-of-the-art video coding algorithms are capable of achieving acceptable qualities at high compression rates (a few kbit/s), suitable for deployment in various applications. In the last few years, wireless applications and Internet have both experienced an increasing popularity. In addition, there has been an increased availability in channel capacity and great progress in digital signal processing power coupled with lower implementation costs. These factors have contributed to the growing interest in the research of low bit rate coding schemes which have brought the visual element into applications with restricted bandwidth, such as mobile communication systems. The most prominent applications which have benefited from high compression video are briefly covered as follows:

- **Video-conferencing** and **video-telephony** provide an audio-visual service which requires low implementation costs and is sensitive to delay and transmission errors. The difference between these two applications is the fact that video-conferencing can both be used in point-to-point communication between two users and also in point-to-multipoint communications involving multiple participants in a session. On the contrary, video-phones are direct point-to-point video transmission applications in which generally only two users are involved. These applications operate over existing wired networks, such as public-switched telephone network (PSTN) and integrated services digital network (ISDN), and also over packet-switched Internet. Due to the limited channel bandwidths of these networks, these applications require low bit rate coding. The constraint on the image resolution is not very stringent since the details need not be accurately displayed.
• Internet-based video is another common application boosted by the growth in Internet usage and the progress in low bit rate video coding.
• Surveillance and remote monitoring can be accomplished by telephone and Internet connections. These applications have become increasingly popular due to the advances in video compression together with low cost, good quality video cameras.
• The term multimedia refers to the integration of a number of services providing full-motion interactive digital video in a desktop environment. Compact disc read-only memories (CDROMs) and digital versatile disks (DVDs) enable the user to interact with the video data. In addition, interactive games and other education and entertainment programs are gaining an ever-increasing popularity. Due to stringent bandwidth restrictions, low bit rate coding is essential to such applications.
• Due to the improvements in video compression, a number of broadcast companies have recently started to supply their subscribers with on-demand services. Such services cover information, such as a variety of news items, and also entertainment.

2.2.1 User Requirements Perspective

The service quality provided by each of the applications mentioned above depends on a number of requirements dictated by the end-user. This sub-section will discuss a number of these requirements, which often have contrasting objectives and improvements in one aspect of the output performance can result in a degradation of another. The aim is to achieve a successful trade-off amongst these partly conflicting factors.

2.2.1.1 Coding Efficiency

Coding efficiency is one of the primary requirements of low bit rate coding. A video image contains a large amount of information and hence, for efficient, low cost storage and transmission of digital video data, substantial compression is required. Current video coding methods exploit both temporal and spatial redundancies present in the raw video signal in order to achieve rate reduction whilst preserving an acceptable output quality. Evidently, the compression rate depends on the video content. For example, a video clip with no high motion activity can be compressed by 100-200 times and still provide adequate quality. The aim of all video coding algorithms is to achieve an efficient compression whilst maintaining adequate quality for a given bandwidth.

2.2.1.2 Perceptual Video Quality

Rate reduction techniques are typically lossy, resulting in a degradation in the reproduced video data. The requirement on the output quality is dictated by the target application. The quality can
be affected by a number of factors including frame rate, number of intensity and colour levels and spatial resolution. Since in the majority of the cases, the decoded information is judged by humans, perceptual quality is one of the major criteria which needs to be optimised. Research has derived formulae which aim to calculate the quality as perceived by the end-user. Amongst them, PSNR as described in Chapter 1, is the most commonly used. Each compression scheme is characterised by a number of parameters which are derived during the encoding process. The perceptual quality measure can be used to optimise these parameters in order to achieve the highest quality. Furthermore, this measure is particularly useful since it allows different coding methodologies to be compared.

2.2.1.3 Delay

The time from when an image is encoded to when it is decoded at the receiver constitutes the end-to-end delay of a video communication system. The end-to-end delay is composed of the encoder delay, channel delay and decoder delay. The encoder delay involves a certain amount of data buffering together with the processing needed for transforming the input data into a compressed bit stream. Channel delay is the time taken for the data to propagate from the transmitter to the receiver whilst the decoder delay depends on the decoding processing time and the sequential arrival of the frames. The processing delay increases proportionally with the amount of motion present in the sequence to be encoded/decoded. Time delays greater than 0.5 seconds are usually perceived as annoying for two-way video communications.

2.2.1.4 Complexity

Complexity is defined by the number of arithmetic computations carried out during the encoding and decoding processes. The computational load in the encoder and decoder depends on the particular application. For example, the decoding process in broadcast applications has more severe requirements since it needs to run in real-time without noticeable delay. The complexity issue is also related to the power consumption. For battery life purposes, the power requirement for mobile applications needs to be low. Due to the great improvements in very large scale integration (VLSI) technology, processing power and affordable memory, real-time and less complex video coders have become a reality.

2.2.1.5 Error Resilience

Efficient compression schemes make use of predictive coding techniques, such as motion compensation. Such methods are highly sensitive to transmission errors due to the fact that the degrading effect of these errors will propagate through the entire sequence. The objective of error resilience is to make the transmitted video sequence more tolerable to the errors introduced into the compressed bitstream and still achieve an acceptable output quality. However, the
improvement in the quality is generally achieved with a degree of redundancy re-added into the bitstream resulting in an inefficient use of the transmission bandwidth. Despite this disadvantage, this property is particularly useful to communications over wireless channels which are characterised by relatively high bit-error-rates (BERs) compared to fixed networks.

### 2.2.1.6 Scaleability

The term scaleability can be defined as a bitstream consisting of a number of embedded discrete layers of quantisation. Each layer corresponds to a particular output bit rate. The decoded quality can suffer from reduced spatial resolution (spatial scaleability), temporal resolution (temporal scaleability) or reduced PSNR (quality scaleability). The output quality can be gradually increased by decoding a higher number of layers at the expense of an increased bit rate. This feature allows the bit rates to scale in order to optimise the quality of the encoded video signal for the given allocation of system resources and channel conditions. In addition, scaleability is obtained without the re-coding of the input signal at the cost of a slight decrease in coding efficiency. The property is particularly useful in a multicast environment, where the receivers might be connected by communication channels of different bandwidths.

### 2.2.1.7 Further Requirements

As interactive video and games become more popular, a friendly user interface and the ability to manipulate data become a key requirement. One particular functionality which is particularly useful is the support for selective video coding, where the user can specify which objects to be coded. Such a feature is provided by the MPEG-4 standard \[N1909, \text{Reade}\].

Moreover, in most video coding applications, video information is accompanied by speech data. Synchronisation, which can be achieved with a number of techniques \[\text{Bryur}\], is thus very important in order to obtain a satisfactory level of performance.

Finally, it is imperative that users who engage in the same communication session are able to access the requested information at the required time, independent of their respective video applications. In addition, it is important to ensure that compressed video data can be decoded at any user end. This requires the encoder and decoder operations to be compatible. As a result, video compression standards should satisfy the interoperability property.

### 2.2.2 Networks Perspective

The objective of a telecommunication system consists of remotely connecting two or more users who might be deploying any of the video applications described in the former sub-section. Apart from providing the transport of compressed information from the source to the required
destination, it is vital that the communication system is also able to handle the routing of traffic amongst multiple active users within a session [Halsal].

PSTN is the traditional analogue telephone network which was designed to carry voice data only. It is a connection-oriented network where the resources are reserved for the whole duration of the connection. This network is characterised by a limited available bandwidth and hence, in some cases, it might not be suitable for providing satisfactory real-time video communication services with adequate perceptual quality and end-to-end delay.

Thus, PSTN was found inadequate for modern communication needs, such as data transmission, facsimile and video. Growing user demands for these services have led to the replacement of this telephone system with an advanced digital system called ISDN. The major difference between the ISDN and the PSTN is the increase in bandwidth availability suitable for supporting the integration of multiple services. ISDN is also capable of supporting a combination of channels interleaved by time division multiplexing. One of the rates that has been standardised supports two 64 kbit/s B channels. Signalling is performed on a separate 16 kbit/s D channel, so that the full 64 kbit/s are available to the user for services, such as video-telephony and video-conferencing. The increase in bit rate is exploited to give a greater throughput resulting in greatly improved video quality and better overall user’s satisfaction. The increasing demand for high bit rate services has been the driving force behind the evolution of broadband-ISDN (B-ISDN) which supports higher transmission rates than ISDN and hence, is more suitable for video communication applications.

Recently, the widespread growth of the Internet has created a mass market for multimedia and information services. As a result, an interesting and rapidly expanding application area for digital video coding can be found in Internet telephony systems. Unlike PSTN, the Internet is a connectionless packet-based network, where the route of packetised compressed video is not defined prior to transmission. Due to the characteristics of transmissions over Internet, the coding schemes employed in such systems are required to be robust to frame or packet losses. In addition, since packets of the same session might be transmitted along different routes, due to network latency and jitter, the packets might not be received in the order that they were transmitted. This requires the video de-packetiser to recover the correct sequence before passing them to the decoder. Both packet loss and out-of-sequence arrival of packets can be detected by checking the packet sequence numbers. However, the re-ordering of packets incurs a delay which is detrimental to real-time services. In the case of a packet loss, it is important for the video decoder to suppress its effect so as to achieve a satisfactory video service on the Internet. One other major problem to be considered is network congestion, causing higher delays which in turn results in a greater
packet loss. Therefore, the video encoder needs to be configured in such a way as to optimise the efficiency of the network by controlling its output rate and hence, avoid congestion.

Finally, video communications over mobile networks are also severely constrained in terms of transmission rates due to the extreme bandwidth limitations. Similar to the Internet, mobile networks are susceptible to channel errors as well as packet losses. In addition, due to the fact that radio is used as the transmission medium, the video stream can further be corrupted by propagation fading, interference, and shadowing in urban areas. Moreover, the mobility property increases further the chances of packet losses. If uncontained, these errors will propagate further through the video sequence rendering its quality unsatisfactory. Therefore, algorithms which provide robustness to channel errors and error concealment techniques are required.

2.3 Low Bit Rate Block-Based Video Coding

Video coding is incorporated by video compression which removes the redundancy from a raw video data stream. Video compression is a process whereby a collection of algorithms and techniques replace the original pixel-related information with more compact descriptions. When compressed data is received over a communication link, it must be possible to expand the data back by decompression. Decompression is the reverse process of encoding, so that it decodes the compressed data back to pixels for display purposes. At its best, video compression is transparent to the end-user. A block diagram of a typical video compression scheme is depicted in Figure 2.1.

![Figure 2.1 A block-based video encoder block diagram](image)

It is now highlighted that the goal of the video compression is to reduce the amount of data required to represent the video signal. This representation is accomplished by performing some or all of the following three steps: redundancy removal from the raw video information, quantisation...
of the remaining data and lossless coding of the quantised values prior to transmission. The existing various video coding algorithms exploit several different video coding techniques, such as waveform-based (discrete cosine transform (DCT) [Rao] and wavelet transform [Katto, Wilki]) coding, object-based coding, model-based coding and fractal-based coding, to achieve these three major steps [Ebrah, Tallu]. This section, however, limits the discussion of video coding methods to the block-based DCT coding technique as this particular method is chosen for the research work. This is due to the fact that block-based DCT is widely employed in international video coding standards, two of which are exploited in this research, namely MPEG-4 and H.263. Furthermore, this method is suitable to fast implementations with low memory requirements whilst achieving an optimal balance between quality distortion and the amount of compression rate.

2.3.1 Block-Based DCT Coding

This is by far the most popular of the current video coding techniques. Most of the existing video coding standards, such as MPEG-1/2/4 and H.261/263/263+/263++ are essentially block-based coders. Block-based coders are quite popular as they make very few assumptions about the input data and hence, work very well across a wide variety of input video signal and across a wide range of bit rates. They mostly make only local references to data (in terms of blocks) and hence, are suitable to fast implementations without the need for large amounts of memory.

Typically, the input video data to these video compression schemes is in YUV colour space in what is called a 4:2:0 format [Rao, Tekal]. In this format, the colour representation for each pixel consists of three components: a luminance (Y) and two chrominance (Cb and Cr) components. HVS is mostly sensitive to the resolution of the luminance component within an image and hence, the Y pixel values are encoded at a higher resolution than the chrominance pixel values. On the other hand, the chrominance values are sub-sampled by two in both the horizontal and vertical directions. This simply reduces the amount of information to be coded by four without significant degradation in visual quality.

The video encoder handles its raw video input on a frame by frame basis. Each individual frame is processed using a particular encoding scheme that has been selected by the encoder [Rao]. If frame encoding is performed using only the data from a single frame with respect to itself, an intra (I) frame is produced. These intra frames allow a video stream to be accessed at random points. On the other hand, inter (P) frames are produced when the content is dependent upon a prediction taken from a certain number of reference frames. These frames are also commonly known as predictive (P) frames and help the video source achieve a substantial amount of transmission rate reduction. Their superior compression efficiency also increases with low motion active video
Chapter 2 Low Bit Rate Video Communication Systems

scenes which is particularly suited to a low bit rate video coding mechanism. All block-based
coders have four essential stages of operation: motion estimation/compensation, DCT coding,
quantisation and entropy coding. The following sub-sections discuss these four operational modes
briefly.

2.3.1.1 Motion Estimation/Compensation

Motion compensation is the stage of the video coding process that reduces the temporal
redundancy of the video signal. Each frame is first divided into rectangular units of 16×16 pixels.
Each of these 16×16 units is known as a macroblock (MB). Each MB is made up of six 8×8
blocks: four luminance and two chrominance blocks (since the chrominance signal is sub-sampled
by two in each direction). In the motion estimation stage, for each of the MBs in the current image
(reference MB), the video encoder searches the previous frame to find an MB location (target
MB) that minimises the pixel intensity differences between the reference and the target MBs. This
particular search is performed within a defined window, named as the search window, around the
target MB, as illustrated in Figure 2.2. The two-dimensional distance of the target MB is known as
the motion vector (MV) of the current MB. Thus, the motion estimation stage computes an MV
for each MB of the current frame, comprising the most complex operation of the whole of the
encoding scheme. The motion compensation stage thereafter applies these estimated MVs to the
previous (reference) image and generates a motion compensated image. Thus, this motion
compensated image (along with the associated MVs) represents an approximation or prediction of
the current image based on the previous (reference) image.

![Figure 2.2 Motion prediction in inter frame coding](image)

2.3.1.2 Discrete Cosine Transform (DCT)

Following the removal of as much temporal redundancy as possible from the video signal by
motion compensation, a DCT is applied to remove the remaining spatial redundancies [Rao]. First
the difference between the original image to be compressed and the predicted image (after motion
compensation) is computed. This is referred to as the residual image or displaced frame difference
signal (DFDS). The DCT is applied to all the 8x8 blocks of the residual image. The DCT converts an 8x8 block of pixel values to an 8x8 matrix of horizontal and vertical frequency components. The DCT de-correlates the spatial information by transforming it into the frequency domain. Most of the energy is then concentrated in the low frequency coefficients. The higher order frequency coefficients (ACs) can now be represented with fewer number of bits without any significant loss in the quality of the reconstructed image.

2.3.1.3 Quantisation

After applying the DCT to the residual image, quantisation is applied to compress the input data. Quantisation maps a range of input values into a single output value. Thus, quantisation comprises the lossy stage of the compression process. The quantised range can concisely be represented as an integer code, which can be used to recover the quantised value during decoding. The difference between the actual value and the quantised value is called the quantisation error or quantisation noise. Each array of 8x8 coefficients produced by the DCT is quantised to produce an 8x8 array of quantised coefficients. Normally, the number of non-zero quantised coefficients is quite small and this is one of the main reasons why the compression scheme works efficiently. Typically, the coefficients are quantised with a uniform quantiser whereby the value of the coefficient is divided by the quantiser step-size and rounded to the nearest whole number to produce the quantised coefficient. The quantiser step-size can be different for different coefficients and may change between MBs. The only exception is the lowest order frequency (DC) coefficient, which is treated differently. The human eye is quite sensitive to large area luminance errors and hence, the accuracy of coding the DC value is fixed.

2.3.1.4 Entropy Coding

Following the quantisation stage, a number of zero valued DCT coefficients exist. Considerable compression gain can therefore be achieved by using a run-length encoding scheme that minimises the redundancy in the binary output. A zig-zag scanning is performed to order the DCT coefficients in a way that clusters most of the non-zero coefficients together. This ordering concentrates the highest spatial frequencies at the end of the scan. These long runs of zeros and non-zero DCT coefficients are then encoded using a variable length coding (VLC) technique, such as Huffman coding [Tekal]. This VLC technique is a lossless coding scheme that assigns short codewords to more probable events and long codewords to less probable ones. On average, the more frequent shorter codewords dominate such that the code string is shorter than the original data.
Following this brief insight into the low bit rate video coding and compression aspects, the next sub-section presents an overview of the major low bit rate video coding standards, which are prominently used for this research work.

### 2.3.2 Low Bit Rate Video Coding Standards

Low bit rate video coding has always drawn significant attention for the past few years. Extensive research potential has been put forward for the investigation of further compression and more efficient coding techniques. These efforts have been initiated due to the existence of successfully implemented video coding standards: H.261 of ITU, MPEG-1 and MPEG-2 of ISO.

Low bit rate coding schemes are needed for all communications media, particularly for those where bandwidth is restricted. Today, the low bit rate video compression and coding algorithm has already been achieved by ITU's internationally well-known H.263 standard. H.263 standard efficiently provides a very low bit rate video coding syntax suitable for a sufficient range of applications, such as video-telephony and video-conferencing via the PSTNs. When the standard was first launched in the year 1996, it was aimed to provide an international standard to insure interoperability between video-phones connected via PSTN which were already in marketplace [Schap].

Apart from the low bit rate support, flexible, efficient and low complexity coding of video has also become a need in particular for today's and future's extensive and bandwidth-hungry multimedia applications over bandwidth-limited networks, such as mobile-wireless channels. Unlike MPEG-1 and MPEG-2, ISO's new MPEG-4 is a content-based coding scheme that also supports very low bit rate wireless mobile multimedia communications and many other functionalities [N1909, Reade].

#### 2.3.2.1 ISO MPEG-4

MPEG-4 algorithm aims to establish a universal, efficient coding of different forms of audio-visual and multimedia data, called audio-visual objects (AVOs) which can be of natural or synthetic origins. Therefore, MPEG-4 is a generic name for a comprehensive multimedia coding and communication standard. However, the discussion of MPEG-4 in this thesis is limited to its visual aspects as a video coding standard. The idea of MPEG-4 was born in need for achieving very low bit rates by high and efficient compression algorithms, particularly for mobile and wireless multimedia communications. Initially, MPEG-4 was targeted primarily at very low bit rate video communications; however, its scope was later expanded to be much more of a multimedia coding standard. The standard does not address any specific application. On the contrary, it supports many clusters of functionalities which may be useful for various applications.
MPEG-4 includes eight functionalities that were not originally supported by existing standards [N1909, Rao, Schap], such as:

- content-based interactivity
  - content-based multimedia data access tools
  - content-based manipulation and bitstream editing
  - hybrid natural and synthetic data coding
  - improved temporal random access
- compression
  - improved coding efficiency
  - coding of multiple concurrent data streams
- universal access
  - robustness in error-prone environments
  - content-based scaleability.

The MPEG-4 architecture allows the separate coding of AVOs, error protection, and the appropriate multiplexing of the separate object elementary streams into a single bitstream together with the scene description information at the encoder side. The transmission may use multiple channels offering various QoS and interactivity levels. At the decoder, the AVOs are demultiplexed, error corrected, decompressed, composited and presented to the end-user [N1909, Puri].

![Figure 2.3 Classification of bit rates and functionalities supported by MPEG-4](image)

The MPEG-4 scheme is based on units of aural, visual or audio-visual content. For this reason, MPEG-4 is accepted to have the content-based functionality which allows a highly flexible access and manipulation of AVOs in the compressed domain. MPEG-4 is efficient across a wide variety
of bit rates ranging from a few kilobits per second (i.e. 5 kbit/s) to tens of megabits per second (i.e. 15 Mbit/s). However, the wide range of transmission bit rate flexibility of MPEG-4 requires a compromise between the supported functionalities and the available channel bandwidth capacity. Consequently, different levels of operational modes are generated: simple profile (low bit rates, i.e. from 5 kbit/s to 384 kbit/s or more, and basic functionalities, such as re-synchronisation and error resilience tools), core profile (moderate bit rates, i.e. from 384 kbit/s to 2 Mbit/s or more, and some core functionalities, such as object-based shape coding, bi-directional prediction (B-frames), temporal scaleability) and main profile (high bit rates, i.e. from 2 Mbit/s to 15 Mbit/s, and distribution and broadcast type of functionalities), as illustrated in Figure 2.3. Unlike in preceding MPEG-1 and MPEG-2 standards where the video information is of a rectangular and fixed size and displayed at fixed intervals, MPEG-4 introduces new video entities, called video object (VO), video object layer (VOL) and video object plane (VOP) lying on top of each other hierarchically [N1909, Puri]. A VO is an individual basic object within a video scene, such as a background or foreground visual object, whereas a VOP is the encoded video object. The encoding and decoding processes in MPEG-4 are carried out on the instances of the VOIs which constitute the VOPs. Lastly, object-based scaleability can be achieved by means of layers known as VOLs which represent either the base layer or enhancement layers of a VOP [Bober, Schaf]. The AVOs can even be of arbitrary shape. The simple profile video coding scheme of MPEG-4 is mainly based on the H.263 video coding algorithm.

2.3.2.2 ITU H.263

The basic configuration of H.263 is an extended version of ITU H.261, and is a hybrid of inter-picture prediction to utilise temporal redundancy and transform coding of the remaining signal to reduce spatial redundancy [H2631, Sadka1]. Thus, the H.263 standard is not radically different from its predecessor H.261. The core structure still consists of block-based DCT coding. It can be thought as an enhanced and optimised version of H.261 scheme with various differences, such as more efficient ways of coding DCT coefficients (with improved adaptive VLC tables), improved entropy encoding and increased precision motion compensation [Yeado]. Moreover, it provides half pixel accuracy, better MV prediction and negotiable operating modes.

H.263 achieves lower bit rates than H.261 (i.e. <64 kbit/s). The coding algorithm is very similar to H.261 in the basic mode, and the options like half pixel MVs, bi-directional prediction with minimal extra delay, arithmetic coding instead of VLC and utilising vectors of adjoining blocks when forming prediction are also supported in the standard.

Both H.263 and MPEG-4 can support various input picture formats (from sub-QCIF: 88x72 pixels to 16CIF: 2816x2304 pixels) and a wide range of frame rates (from 5 fr/s to 30 fr/s). Furthermore,
both H.263 and MPEG-4 are equipped with a number of certain negotiable options, two of which are worth discussing in brief detail in this chapter as they are utilised during numerous experiments conducted in this thesis work. These negotiable options are namely the unrestricted MV mode and the advanced prediction mode. The term negotiable is commonly used for these and the other optional modes since the decoder can signal to the encoder which option it has the capability to decode prior to the actual data transfer. As a result, the encoder proceeds to use the confirmed optional modes, provided that it also has the capability. The use of these options makes MPEG-4 and H.263 very flexible in terms of interoperability in heterogeneous environments.

2.3.2.3 Unrestricted MV Mode (Embedded in MPEG-4, Annex D of H.263)

The default video source coding algorithm restricts the MVs such that all pixels referenced are within the coded picture area of the previous reference frame. However, when this mode is used, this restriction is no longer valid and MVs are allowed to point outside the picture. The benefit of this mode can easily be recognised if the following situation is considered. Assuming that one column of new pixels move into the picture then this means that all pixels except this column of new pixels can theoretically be predicted from the previous picture. However, the restriction on the MVs in default mode implies that those blocks near the edge of the picture will suffer from poor prediction due to the one pixel motion. Conversely, the MVs are allowed to point outside the picture and the referenced pixels outside the picture are replaced by the nearest edge pixel with the use of this option. Therefore, it is possible to achieve a good prediction for all pixels except for those which are newly introduced to the scene. This option gives particularly better results when there is high camera movement.

Moreover, the option deals with an extension to the overall range of the MVs. In default prediction mode, the values for the MVs are restricted to the range \([-16.0, +15.5]\) pixels. With this option, the maximum range for the MVs is extended to \([-31.5, +31.5]\) pixels. However, it should be noted that not all the vectors may be reached at any time. If the prediction is in the range \([-15.5, 16.0]\), only the values that are within a range of \([-16.0, +15.5]\) around the predictor can be reached. In addition, if the predictor is outside \([-15.5, 16.0]\), all vectors within the range \([-31.5, +31.5]\) can also be reached. For obvious reasons, the gain is negligible for a static camera and low activity picture but is particularly useful when there is a large object or camera motion.

2.3.2.4 Advanced Prediction Mode (Embedded in MPEG-4, Annex F of H.263)

This option includes the possibility of using four MVs instead of one per MB. If this mode is used, then the unrestricted MV mode must also be switched on to access the pixels which are located outside the normal coded picture area.
Chapter 2 Low Bit Rate Video Communication Systems

In default operation of a video encoder, a 16x16 block is used for motion compensation. However, advanced prediction mode uses an 8x8 block instead. Consequently, this may provide more accurate prediction but may, in some cases, also impose an additional overhead for coding of the four MVs. A trade-off between bit rate and quality has to be established. This can be decided on an MB-by-MB basis if there is sufficient benefit to use four MVs instead of one. However, in most cases, the output rate may also reduce with the use of these two modes of operation. This is due to the fact that in high motion areas, a better motion compensation for the newly established four MVs is achieved. This will be demonstrated in the computer simulations conducted in the fifth section. As in normal operation, each component of the four MVs is differentially encoded. The predictors are calculated separately for each of the horizontal and vertical components, as shown in Figure 2.4.

2.4 Error Resilience in Video Communications

The prominent area of interest of low bit rate video communications is the wireless networks where the receiver device is constrained in power which also limits the available bandwidth. Additionally, the number of users imposes further limitations on the scarce resources, in terms of transmission bandwidths. Apart from the channel bandwidth constraints, wireless multimedia communication channels are also characterised by varying delays (jitter effect) and high BERs. These effects can have serious destructive impacts on the picture quality of video. Moreover, in a long round-trip delay environment, like a satellite link, the synchronisation of a video encoder and decoder is severely destroyed. This fact causes an accumulation of errors and a loss of correlation between the transmitted and received frames.

Due to the rapid growth of mobile communications, it is extremely important that access is available to audio, video, speech, text and graphics multimedia information via wireless networks. This, basically, implies a need for useful operation of multimedia compression algorithms in error-
prone environments at low bit rates. Mobile-wireless channels are typically noisy and communications over these channels suffer from a number of channel degradations due to random and bursty errors caused by interference, fading and multipath reflections [Saund, Sklar]. The effects of these channel errors, particularly on compressed video can be very severe [MacDo, Tallu2]. VLC renders video communications more vulnerable to these channel errors. Similarly, predictive coding, which highly depends on the previously encoded video frames, also makes video transmissions quite susceptible to error-prone environments. Both coding algorithms are widely used in low bit rate block-based video coders which are employed in wireless multimedia communication channels where bandwidth is remarkably limited and therefore, expensive. Erroneous VLC data causes synchronisation losses between the encoder and decoder whilst error in predictive coded data causes rapid propagation of channel errors both spatially and temporally through the entire video sequence causing severe quality degradations [Dogan1, Kawah, Sadka1, Tallu2]. Lack of attempts to stop both error effects, either by the encoder or the decoder, may result in the entire loss of the video transmission.

Therefore, multimedia video compression algorithms should take strict actions against the error effects in the transmission media. These responses can be achieved by encoding the video data error-resilient to channel errors and fades. There are several different methods that can be exploited, such as the provision of a higher transmission power or the use of automatic repeat request (ARQ) protocols [Halsa2]. The former requires excess amount of transmission power to cope with poor channel conditions. However, this forms a very expensive and crude approach to the problem and does not mitigate the error effects directly. The latter is also discouraged by longer waiting latencies of the return signal in case of a feedback algorithm and can be employed by the use of a threshold which is a function of both the number of errors and the round-trip delay.

The most suitable solutions seem to be the utilisation of forward error correction (FEC) techniques [Berro, Forne, Sween, Wicke] and/or error resilience algorithms [Dogan1]. H.263+, H.263++ and in particular MPEG-4 address various error resilience functionalities which provide robustness in error-prone environments. Before discussing the error resilience approach in further detail, it is worthwhile mentioning the different types of errors in the transmitted video stream and their effects on the QoS of the video communications.

### 2.4.1 Types of Errors in Video Transmission

Mainly, four types of errors can be observed in an erroneous video transmission [Sadka2]. Although each of the error types is individually described in this sub-section, in reality, different types can occur simultaneously. The first type causes the least significant data loss. For instance, the errors in the wireless communication environment can cause some loss in the texture data of a
video frame. This loss might take place only in any MB of the still background scene of the frame, which does not have any motion data at all. Consequently, a single bit error on one of the parameters does not have any influence on the segments of data other than the affected parameter. In this case, the error is limited to a single MB that does not take part in any further reconstruction process due to the lack of motion. Thus, any kind of temporal error propagation from one frame to another cannot be expected. Similarly, this kind of one background MB error does not cause any loss of synchronisation either, which results in a spatial error propagation between MBs within a frame. Thus, the damage is localised and confined only to the affected MB without affecting any subsequent data. This kind of error in the video stream is the least destructive one as it does not cause any loss of synchronisation of the decoder with the encoder. However, it still renders the video frame fairly degraded in picture quality.

The second type of error results in a prediction loss. This type of error is more problematic as it incurs an accumulative damage both in time and space due to prediction. The error propagates to subsequent predictive (P) coded frames due to the temporal dependency induced by the motion estimation and compensation processes. This type of error does not cause any loss of synchronisation either as the decoder is able to skip the right number of bits of the erroneous codewords.

The third type of error gives rise to a synchronisation loss. In this case, the decoder becomes totally unaware of what part of a frame the received information belongs to. In this circumstance, when the decoder detects an error in a VLC, it skips all the forthcoming bits, regardless of their correctness. The skipping continues until the next error-free re-synchronisation word is detected in the video stream. Since the decoder discards even all the useful and error-free information it receives between those two synchronisation points, this process transforms a single bit error corruption into a burst of errors.

Although VLCs provide good compression ratios, they also cause the transmission errors to propagate quickly in the video bitstream. Errors are generally detected when illegal VLCs are found in the bitstream, but in some cases channel errors result in valid codeword entries in the VLC Huffman tables. In these situations, the decoder continues to decode these erroneous codewords without realising they contain incorrect information. It takes some time for the decoder to detect the error until it reaches a stage where the following data is realised to be an illegal VLC. At this point, the decoder has to re-synchronise itself to the remaining VLC data. This type of error causes one DCT based function to dominate the appearance of the block, which is highly perceivable as a distortion in the image. Another very noticeable type of distortion is caused by incorrect decoding of the DC component of a chrominance block which tends to cause the colour
of a 16x16 block to be predominantly pink, cyan or green. Eliminating both of these types of distortion significantly improves the perceived image quality [Lee1].

Finally, the fourth type of error is the worst of all the four kinds as it comprises header data losses within the video frames due to severe error conditions. In this case, the decoder can no longer follow the encoder at all. This incident might sometimes result in discarding and hence, losing an entire video frame even though the rest of the frame is received correctly.

2.4.2 Error Resilience

Research in video communications does not only focus on a number of compression and coding algorithms, but it also includes various reliable transmission aspects of the visual information from a source to a destination point. Error protection and recovery schemes are the most significant and challenging research and development areas of the video communications. The reason is that compressed visual information is highly susceptible to transmission channel errors due to the VLC and predictive coding nature of the video coding algorithms, as previously discussed. Therefore, it is important that the video data is encoded in an error-resilient way at the source. Error resilience is the most widely exploited method for the protection of video data prior to its transmission. Resilience is a measure of the ability of a video source to adapt itself to varying channel conditions. Resilience might imply some significant changes in the transmission order of video data as well as its structure to render the video stream more robust to communication channel errors. There are many different error resilience methods and algorithms which have been designed [N1646]. Principally, error resilience is applied to provide robustness against the destructive error effects and prevent the error propagation throughout the received video stream. Error resilience methods can be classified into the following four major categories [Lee1]:

2.4.2.1 Error Detection

In a typical block-based video compression technique that uses motion compensation and DCT, such as an MPEG or H.26X group video compression algorithm, the following checks are applied to detect the bitstream errors:

- The MVs are out of range.
- An invalid VLC table entry is found.
- The DC coefficient is out of range.
- The number of AC coefficients in a block exceeds 63 (64th is the DC coefficient in an 8x8-pixel block).
When the decoder identifies any of the above conditions in the process of decoding a video bitstream, it flags an error and jumps to the error handling procedure. However, due to the nature of the video compression algorithms, the location in the bitstream where the decoder detects an error is not always the same location where the error has actually occurred but some undeterministic distance away from it, as observed from Figure 2.5. As a result, once the decoder detects such an error, it loses synchronisation with the encoder.

2.4.2.2 Re-Synchronisation and Localisation

In some cases, channel errors can cause loss of synchronisation at the receiver. The loss of synchronisation may be caused by insertion or deletion losses due to the instability of the clock. Bit errors can also cause loss of synchronisation when VLCs are used. When compression schemes with memory are used, errors caused by synchronisation losses are compounded by the propagation of errors after synchronisation is regained.

One way of introducing resilience to this type of errors is by attempting to detect and localise the error. This is usually done by including unique re-synchronisation codewords. These particular codewords are usually quite long and are hence infrequently used. For coding schemes with memory even the inclusion of re-synchronisation words may not be sufficient alone and hence, in some cases, the insertion of periodic intra (I) frames into the video sequence may also be required.

The use of synchronisation codes and periodic restarting is sufficient to prevent catastrophic losses for all types of channel errors. Certain measures have also been introduced to target errors caused by using VLCs. VLCs allow the propagation of errors since errors in a codeword can cause a loss of synchronisation. A simple way to prevent this is to use fixed length codewords (FLCs), yet at the expense of a possible coding efficiency degradation. It is also possible to re-arrange the VLCs in the bitstream in such a way that the start points of the codewords are also known to the decoder [Redmi].

Moreover, re-synchronisation enables the use of efficient error resilience schemes, such as two-way decoding with the use of reversible VLCs (RVLCs) [Sadka1, Watan]. These are special codewords which allow the receiver to decode an erroneous bitstream forward and backwards.
once the re-synchronisation is established to salvage as much useful information as possible for an acceptable video QoS [Dogan1].

2.4.2.3 Unequal Error Protection

In video coding, some information is more important than others. For instance, a loss of information about quantisation levels or MVs is more damaging than a loss of information about DCT coefficients. Even for DCT coefficients, the DC coefficient and low frequency coefficients contribute more to the subjective and objective quality of video than the high frequency coefficients. Thus, when the capacity of the channel is limited, it is sensible to give different levels of protection to data according to its importance.

This feature can be provided by scalable coding where the bitstream is divided into several layers. The quality of received video increases as information from successively higher layers is received. The information from the lower layers is usually more important and hence, can be given higher protection. Where scalability is not appropriate or too complex to employ, video stream can be divided into different portions of different importance levels, as in the data partitioning mode of MPEG-4 [Budag, MPEG4, Tallu2]. Then, each portion can be given a separate and hierarchical priority. For instance, the most powerful protection is applied to the headers and administrative data, followed by the MVs. Finally, the DCTs are given the weakest level of protection.

2.4.2.4 Error Concealment

When information is lost due to transmission errors, entire 8×8-pixel blocks may be missing from the decoded image. A good approximation of these lost blocks can often be obtained by using information already available at the receiver. The simplest method is to replace the block with the block from the same position in the previous frame. An estimate of the MV of the lost block can also be made using the MVs of neighbouring blocks. This particular operation comprises the most straightforward error concealment algorithm which provides very acceptable results for low bit rate, low delay applications. More elaborate techniques have also been proposed and presented in the literature [Shira, Wang]. The effectiveness of a concealment strategy is highly dependent on the performance of the successful re-synchronisation and the correct localisation of the errors in a video stream.

Error resilience techniques are numerous. However, two of them will be the primary concern of this thesis as they are effectively used for the research work presented here in particular. These two techniques, namely the feedback control signalling (FCS) and adaptive intra refresh (AIR) methods, will further be discussed in the following sub-sections in detail. The reason for the choice of these two methods of error resilience provision is that they particularly give reasonable
amount of robustness to video transmissions in both random and bursty error-prone video communication environments, as will be demonstrated in the forthcoming parts of this thesis. Moreover, both of the techniques do not require any significant modifications of the decoder operations. This particular feature is supported by their standards-compliant operations which support their flexible application-independent utilisation.

2.4.3 Feedback Control Signalling (FCS)

FCS algorithm is an adoption of Annex N: reference picture selection mode of the H.263+ standard which relies on a return channel signal informing the source coder of the lost, corrupted or properly delivered video frames. The encoder, which gets a backward channel signal from the decoder, uses only the correctly decoded part(s) for the prediction of an inter (P) coded frame. Thus, this prevents a temporal error propagation throughout the video sequence and improves the picture quality in error-prone environments where the delay factor is also significant. Considering two-way communications, FCS shows a synchronised pattern of encoder and decoder operations without insertion of any intra (I) frames. Practically, this FCS operation inhibits the massive increases in the transmission bit rate and hence, the waste of invaluable bandwidth conventionally induced by the frequent I-frame insertions. The FCS algorithm reduces the error accumulation which has a severe impact on video communications. Therefore, FCS is a very convenient way of countering packet or frame losses during transmissions [Megge]. In this way, the reference picture selection and the long-term prediction operations are accomplished by the source encoder.

The algorithm is based on an acknowledgement (ACK)/non-acknowledgement (NACK) signal or both signals together which return back to the encoder by the use of a reverse channel. Thus, in this technique, the reference picture for a P-frame in the encoder is replaced adaptively depending on the feedback signal from the receiver. As a result, both the encoder and the decoder operate interactively. The encoder has two kinds of operating modes: the ACK mode and the NACK mode. The encoder switches these modes according to the channel error conditions [N1646, H2632]. In this technique, unlike in default video coding schemes, where only the last locally decoded and reconstructed frame is stored, the encoder also stores more than one previously decoded and reconstructed frame in the local encoder buffer. The size of the local buffer can dynamically be changed according to the duration of the round-trip delay of the received feedback signal. The round-trip delay is from the time when a source transmits a video frame to the time when it receives an ACK/NACK signal of the corresponding frame. Figure 2.6 depicts an encoder FCS operation for a 2-frame round-trip delay scenario.
In such a scenario, the encoder buffer updates itself every time an ACK message is received, and drops the previously decoded and acknowledged locally reconstructed frames sequentially. Similarly, decoder buffer drops the stored preceding frame when it receives a new correct frame and acknowledges that particular frame. In case of not receiving the expected frame, the decoder freezes and displays the former correctly received frame in its buffer until a new video frame is received without any errors.

2.4.4 Adaptive Intra Refresh (AIR)

The definition and the detailed operation of AIR are discussed in Annex-E.1.5 of the MPEG-4 visual standard [MPEG4]. The AIR method is applied to increase the error robustness of the rather sensitive portions of the video data against varying network delays and high BER conditions over error-prone communication channels. It maintains a certain quality in diverse channel environments with different error characteristics. In this method, the encoder evaluates and detects highly vulnerable portions of the video stream, such as high motion MBs. This information is then encoded with respect to itself using intra (I) mode, without any prediction. This evaluation is carried out by comparing the sum of the absolute difference (SAD) [Rao, Tekal] and the SAD threshold (SAD_th) values in those particular MBs. SAD is calculated between the current MB and the corresponding MB in the previous frame whilst SAD_th is a threshold limit which represents the average SAD value of the entire MBs in the previous frame. Since the SAD values are already computed during the motion estimation of MBs for a video frame, the AIR algorithm does not impose any additional complexity burden on the entire coding scheme. If a particular SAD value exceeds the certain threshold SAD (SAD_th), the encoder deduces that the particular MB belongs to a high motion area which is regarded as susceptible to channel errors. In block-based video coding, video frames consist of rows of sequential MBs. Therefore, in order to
separately locate the I-MBs in different video packets, the detected moving areas are scanned vertically and a certain number of MBs are decided to be coded in intra (I) mode instead of inter (P) mode. By spreading the I-MBs into different video packets, the loss of these MBs is minimised in cases of packet losses. The operation of the AIR algorithm for a QCIF (176x144 pixels) video frame is depicted in Figure 2.7 where the grey area represents the moving object.

Therefore, the main aim of the technique is to protect the high motion areas from transmission errors, due to the fact that high motion areas are more susceptible to channel errors as they are more difficult to be concealed. Any corruption in these particular areas can propagate very quickly. The major distinctive feature of the AIR algorithm is that it tracks the high motion areas of the video scenes within the frames and adaptively encodes them in I-mode. This protection is provided by the I-mode of the encoding scheme which afterwards helps the end-decoder refresh its memory. In this way, the detrimental effects of propagating errors within the entire video sequence is stopped. This approach incurs a negligible encoder complexity and imposes no significant modification in the decoder. However, this is accomplished at the expense of increased transmission bit rates. Nevertheless, the AIR method still gives much less bit rate expansions compared to the conventional method of frequent insertion of full-scale I-frames to refresh a video stream. Thus, AIR smoothes out the undesirable peaks in the output rate, usually obtained by the full I-frame insertions, by only refreshing the area of interest in the active scenes.

### 2.5 Source Material

To evaluate the performance of the developed techniques and algorithms throughout the thesis, various ITU video test sequences with different properties are used. "Akiyo_with_Crowd", "Carphone", "Claire", "Container", "Foreman", "Miss_America", "Mother & Daughter", "Salesman", "Sign_Irene" and "Suzie" are the ten video sequences that are tested, as illustrated in
Figure 2.8. The reason for choosing such video clips is to demonstrate the test results on a prevalent platform that is also used by the other video coding and communication researchers and experts around the world. In this way, the results can be compared universally.

![Figure 2.8 The original first frames of the video test sequences used in the thesis:](image)


The video test sequences are utilised during the experiments in their coloured natures with QCIF sizes containing 176x144 pixels. The entire video clips can be divided into three major categories depending on their motion active scenes. The motion activity and the amount of detailed features within a video sequence directly affect the number of bits required to encode that particular sequence. Therefore, high motion active and very much detailed images are rendered high bit rate video clips. Amongst the ten clips, "Akiyo_with_Crowd", "Sign_Irene" and "Foreman" comprise the highest motion activities. Thus, these three form the first category: high motion scenes. "Akiyo_with_Crowd" is a video clip of a news reader with a high motion background where new objects enter and leave the scene continuously whereas "Sign_Irene" is a video sequence of a person using high motion sign language to communicate with her audience. Both video clips comprise very detailed background images with full colour features. On the other hand, "Foreman" is a video sequence which includes a talking person with a complex and shaky background whereby the compounded camera motion (panning and tilting) makes it very difficult to work with. Therefore, in most of the experiments throughout the thesis, "Foreman" is used as to show the effectiveness of the developed algorithms since it is believed that any scheme resulting in acceptable qualities with "Foreman" will also work fine for other sequences. The second category comprises moderate motion scenes, which includes "Carphone", "Mother & Daughter", "Salesman" and "Suzie". "Carphone" shows a talking person on a moving background with fair details whilst "Mother & Daughter" consists of two persons in front of a stationary background. "Salesman" contains a talking person with fast hand and arm motions and a very detailed stationary background. "Suzie" presents a very fast head motion in the middle of the sequence.
which renders this particular clip a moderate motion active scene even though it is a talking head-and-shoulders type of image on a stationary background. And finally, the remaining “Container”, “Claire” and “Miss_America” sequences fall into the third category: low motion scenes. “Container” comprises a slowly cruising container ship and the associated slow camera panning. “Claire” and Miss_America” are another talking persons type of video clips with stationary backgrounds. Moreover, “Claire” contains a rather colourful background compared to the very dark background of “Miss_America” which is usually more suitable for video object segmentation purposes. Apart from the QCIF size “Container” sequence, all the other QCIF video clips are head-and-shoulders type of test sequences which are genuinely believed to be the primary interest on 3G mobile communication devices with limited power and display sizes.

2.6 Computer Simulations and Analysis of the Results

Before moving further into the video transcoding world, it is believed to be very helpful to conduct a few pure low bit rate video coding and error resilience tests for a more lucid understanding of the forthcoming contents and the relevant discussions throughout the thesis. Therefore, this section presents the preliminary experimentation on the behaviours of the two low bit rate video coding standards, namely H.263 and MPEG-4. In addition, more experiments are also performed and their results are demonstrated for the AIR and FCS error resilience algorithms when they are directly applied to the source coding schemes. The experiments are performed by means of computer simulations. The test set-ups and the simulation conditions are clearly described in each of the individual experiment.

2.6.1 MPEG-4 and H.263 Tests

2.6.1.1 Experiments and Results

In this part of the preliminary experiments, an H.263 and an MPEG-4 video encoder and decoder sets are tested. For this purpose, two different video test sequences were encoded and decoded in compliance with the H.263 and MPEG-4 video coding standards. These two sequences are the 150-frame “Suzie” and 200-frame “Foreman” video clips. Both clips were encoded at a frame rate of 25 fr/s, with QCIF (176×144 pixels) size and in I-P-P-P-P-... format.

The first set of the test results comprises the performance evaluation experiments on the H.263 negotiable options, namely the unrestricted MV mode and the advanced prediction mode. These results are depicted in Figure 2.9. During the simulations of “Suzie”, the quantisation parameter
(QP) was kept constant at 10 whereas different QPs were tested for “Foreman” to observe the effects of the quality parameter variation on the video quality.

Furthermore, Figure 2.10 demonstrates the second set of the results. These results present the performance comparisons made between the H.263 and MPEG-4 operations at similar bit rates with the use of the two negotiable options. The objective results are also summarised in Table 2.1.

Finally, Figures 2.11-12 depict the subjective results obtained from the last frames of the 150-frame “Suzie” and 200-frame “Foreman” sequences, respectively.

Figure 2.9 The objective results of H.263 direct enc/dec a- “Suzie” and b- “Foreman” with and without options

Figure 2.10 The comparative objective results of H.263 and MPEG-4 direct enc/dec a- “Suzie” and b- “Foreman” (with options)
Figure 2.11 The subjective results of the 150th frames of “Suzie”, a- original; b- H.263 direct enc/dec (without options); c- H.263 direct enc/dec (with options); d- MPEG-4 direct enc/dec (with options)

Figure 2.12 The subjective results of the 200th frames of “Foreman”. Top row- H.263 w/out options; middle row- H.263 with options; bottom row- MPEG-4 with options. First column- QP=4; second column- QP=6; third column- QP=11; fourth column- QP=20
Chapter 2  

Low Bit Rate Video Communication Systems

<table>
<thead>
<tr>
<th>QP</th>
<th>H.263 w/out options</th>
<th>H.263 with options</th>
<th>MPEG-4 with options</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>55.520</td>
<td>35.620</td>
<td>52.650</td>
</tr>
<tr>
<td>200-frame “Suzie”</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>347.017</td>
<td>37.883</td>
<td>308.475</td>
</tr>
<tr>
<td>6</td>
<td>198.525</td>
<td>35.610</td>
<td>172.608</td>
</tr>
<tr>
<td>11</td>
<td>86.550</td>
<td>32.750</td>
<td>75.890</td>
</tr>
<tr>
<td>20</td>
<td>50.090</td>
<td>30.092</td>
<td>47.120</td>
</tr>
</tbody>
</table>

Table 2.1 The summary of the objective results of “Suzie” and “Foreman”

2.6.1.2 Analysis of the Results

As illustrated in Figure 2.9 and Figures 2.11-12, the H.263 video quality increases with the use of the two negotiable options. The average improvements throughout the sequences consist of 0.1 dB and 0.2 dB for “Suzie” and “Foreman”, respectively. The reason why “Foreman” showed a higher performance than “Suzie” is that the motion activity is also higher in this particular sequence. The use of advanced prediction mode allows more MBs to be encoded with four MV sets as opposed to the conventional one MV per MB encoding. Consequently, this situation results in a better motion estimation and compensation in these particularly high motion regions. Similarly, “Suzie” also presents much better quality level when there is the high motion region, which indeed falls into the middle part of the sequence. Here, the instantaneous quality improvement has been observed to be more than 1 dB, as depicted in Figure 2.9.a. Conversely, low motion regions present similar quality levels for both cases when the options are used and not used. Evidently, compared to the quality levels of “Foreman”, the overall performance improvement reduces when averaged over the 150 encoded frames. On the contrary, “Foreman” results show continuous increased PSNRs in favour for the use of negotiable options. Nevertheless, using the options still gives better qualities than encoding without the options for both of the test clips. In addition to the contributions of the four MVs to the better performance in high motion areas, the higher quality is also due to the fact that the use of unrestricted MV mode increases the quality where there is high camera motion, such as in “Foreman”, and when there are new objects coming into the scene, such as in both “Foreman” and “Suzie”.

The output bit rate was expected to increase with the negotiable options due to the use of four MVs per one MB rather than one MV. However, it is not the practical observation in most cases. Similarly, “Suzie” and “Foreman” also presented reduced bit rates when the options were on, as seen in Table 2.1. This is due to the fact that the motion-related options not only increase the quality, but also provide a better motion compensation for high motion regions. This can easily result in lower rate outputs, as also observed here.
Furthermore, the performance of MPEG-4 video has been noticed to be slightly better than that of H.263 at the same QPs, as observed from Figure 2.10. This is due to the fact that the MPEG-4 video coding standard uses slightly higher number of bits than the H.263 standard to encode a video stream at similar operating conditions. Moreover, the MPEG-4 video coding standard uses two Huffman tables as opposed the only one table of the H.263 standard which results in a better representation of the pixel values within a scene. Thus, the higher number of the Huffman VLC codewords, the better the picture estimation in terms of quality. This slightly better quality has been perceived to be valid for both of the test sequences. The slight difference has been recorded as maximum 0.2 dB in Table 2.1. However, the maximum of 0.2 dB better quality can only be observed in the objective results whereas this minor quality difference cannot be distinguished in the subjective qualities depicted in Figures 2.11-12. Naturally, it has also been observed in Table 2.1 that the small increase in the picture qualities were obtained at the expense of small increase in the bit rates. The small variations in the bit rates are due to the variable bit rate operations of the encoders. Since the resulting fixed quality was the primary parameter to be observed whilst conducting the experiments, the output bit rate was allowed to vary. Therefore, these small variations are acceptable whilst comparing the quality results of the H.263 and MPEG-4 simulations.

Finally, it has also been demonstrated that increasing QP decreases the output bit rate (Table 2.1) and also degrades the video quality due to the coarser quantisation levels, as depicted in Figure 2.9.b, Figure 2.10.b, Figure 2.12 and Table 2.1. The reason behind this observation is that the increase in QP levels results in more zero DCT coefficients which further reduce the output bit rate. However, rate reduction occurs at the expense of a quality degradation as coarser quantisation causes blurry pictures. Here, the quality loss between the lowest and the highest QPs has been observed to be around 8 dB for “Foreman” for a substantial amount of rate reduction, which was around 260 kbit/s. Thus, the overall rate reduction achieved for a QP difference of 16 is 85%.

2.6.2 FCS Algorithm

2.6.2.1 Experiments and Results

To perform the FCS experiments, an MPEG-4 video encoder in the baseline mode was modified and used for the computer simulations. However, before the resilience experiments, the performances of various video test sequences in error-prone channel conditions were also investigated. With a simple algorithm, randomly chosen encoded video frames were discarded from the entire video sequence to simulate frame dropping effects in error-prone environments. Then, the resilience experiments were carried out.
In the experiments, three different video sequences with different motion activities were used: "Foreman", a highly active sequence; "Container" and "Claire" with moderate activity. All three of the test sequences were encoded in I-P-P-P-P-... format at 25 fr/s frame rate and for 200 frames. Frame sizes were QCIF (176×144 pixels) and quantiser step-sizes were kept constant at 10 throughout the encoding processes without any rate controlling algorithm.

The computer simulation results are demonstrated in Figures 2.13-14 and Table 2.2 for objective and subjective measures, respectively. The following sub-section presents the discussions of the obtained results for the 3.5% frame loss transmissions. Seven randomly dropped frames are indicated in the captions of each figure. The simulation results are presented for error-free, non-resilient error-prone, error-resilient with 1-frame delay and 2-frame delay operations in these figures and the relevant table.
Figure 2.13 The objective results (the lost frame numbers: 6, 22, 62, 81, 108, 127 and 188)

a- error-free sequences

b- non-resilient error-prone sequences (3.5% frame loss)

c- 1-frame delay error-resilient sequences (3.5% frame loss)
2.6.2.2 Analysis of the Results

The simulation results are presented in two separate figures. Figure 2.13 depicts the objective results, in terms of PSNR and bit rate variations, whilst Figure 2.14 illustrates the corresponding subjective results of the 200th snap-shots of the test sequences, respectively. The left-hand side column of Figure 2.13 shows PSNR performances for each of the three video sequences, and the right-hand side column presents the bit rate graphs for the corresponding sequences.

As observed from the first column graphs of Figure 2.13, there are significant degradations in the PSNR values of the non-resilient error-prone sequences with respect to the PSNRs of the error-free sequences. In particular, the PSNR degradation in the “Foreman” sequence is notable in contrast with the other two sequences as the frame losses affected a very highly active scene. High motion activity can also be noticed by studying the bit rate versus frame number graph of “Foreman”. In this particular graph, bit rate variation fluctuates much more than the ones of the other two sequences. Furthermore, the average bit rate is also much higher than the others. The dashed lines show the error-resilient sequences with 1-frame delay case and the dotted lines show the error-resilient sequences with 2-frame delay case in both column graphs. As expected, there are severe PSNR drops in the objective results which correspond to the dropped frames. However, the performance improves, mostly reaching the error-free levels quite swiftly. On the other hand, the error-prone sequences present continuous drops as these are not resilient to frame losses. This is due to the fact that the prediction is taken from erroneously reconstructed frames. However, in the error-resilient cases, encoder is informed back of the error conditions and it immediately stops

Table 2.2 Average PSNR and bit rate values of the three test sequences
taking prediction from the erroneous frames notified by the decoder via the feedback signal. Thus, FCS technique ceases error accumulation which has severe impacts on two-way video communications.

The depths and the widths of the sharp PSNR falls in the resilient video sequences are directly related to the round-trip delays of the feedback signals. Since the 2-frame delay error-resilient encoder waits more time than the 1-frame delay error-resilient one, the drops in its PSNR values happen to be deeper and wider than the other ones. This is due to the reason that the 2-frame delay resilient encoder takes the prediction of a currently encoded resilient frame from a much earlier frame in its buffer. Inevitably, this causes less correlation between the currently encoded frame and the previously reconstructed and stored frame, and their corresponding MBs. However, in the 1-frame delay resilient case, the correlation of those two frames is higher due to the lower latency of the feedback signal. The widths of the sharp drops clearly depict the delay duration, in terms of number of frames; wider for the 2-frame delay case and narrower for the 1-frame delay case, respectively. Different levels of scene changes in the video sequences cause varying PSNR drops. Thus, the differences in the levels of sharp performance drops can also be observed in the figures.

The right-hand side column of Figure 2.13 shows the bit rate variations of each of the three test sequences. Here, as opposed to the sharp falls of the resilient sequences in the left-hand side graphs, a number of different size spikes are seen. These spikes correspond to the frame numbers which were encoded in the resilient way. Therefore, the number of bits to encode these particular frames appear to be higher in amount than the conventionally encoded ones. Naturally, this increases the bit rate of those particular frames. Moreover, the whole average bit rate of the entire session also increases. The spikes of the 2-frame delay resilient cases are generally higher than the ones of the 1-frame delay cases. The reason is that the lack of correlation between the frames during the resilient encoding results in coding of those resilient frames with higher number of bits. This observation is generally valid. However, in some cases, the spikes of the 1-frame delay resilient bit rates are higher than the others. This can be explained by the different levels of scene changes throughout the sequence.

Performance evaluation, in terms of PSNR levels, can also be accomplished by the study of Table 2.2. The table demonstrates that the error-resilient performances are better than the non-resilient error-prone operations for all three of the tests. Moreover, it can be observed that the resilience performances are slightly better when the round-trip waiting latency is shorter. Therefore, the 1-frame delay resilient operations show better qualities than the 2-frame delay resilient cases. This is due to the higher amount of correlation between the current and reference pictures, as discussed earlier. It can also be observed that the average bit rates of the non-error-resilient (both error-free and error-prone) sequences are the least of all, and the bit rates increase as the delay increases in
the error-resilient sequences, as demonstrated in Table 2.2. The highest bit rates correspond to the 2-frame delay resilient sequences as the longer the delay the less correlated the frames during the prediction process. Thus, this event also increases the number of bits required to encode these particular frames.

Finally, Figure 2.14 depicts the 200th frames of each sequence for the subjective quality assessments. All three of the sequences perform rather degraded picture qualities for the error-prone cases, but better qualities for the resilient ones. Furthermore, the picture qualities of the 1-frame delay resilient sequences are also slightly better than the ones of the 2-frame delay resilient sequences due to the reduced delays and increased correlation between frames. The degradations or the improvements might seem to be imperceptible for “Claire” and “Container” sequences due to their less motion activities. Conversely, the changes in quality are more discernible for the high motion “Foreman” sequence.

### 2.6.3 AIR Algorithm

#### 2.6.3.1 Experiments and Results

Lastly, this sub-section provides the test results and the discussions of the second error resilience method, namely AIR, which is used for robust video communications in error-prone environments throughout the thesis. For referencing purposes in the forthcoming chapters, the resilience algorithm is directly applied to the source coding MPEG-4 scheme here. Figures 2.15-17 and Table 2.3 present the objective and subjective quality results.

![Graphs showing subjective results](image)

**Figure 2.15** The subjective results of (a) “Suzie” and (b) “Miss_America” (BER = 1e-03)
Chapter 2  

**Figure 2.16** The subjective results of the 150th frames of “Suzie” over a BER = 1e-03 random error channel, a- error-free; b- non-resilient error-prone and c- error-resilient with AIR

<table>
<thead>
<tr>
<th>Type of the Scheme</th>
<th>Av. Bit Rate [kbit/s]</th>
<th>Av. PSNR [dB]</th>
</tr>
</thead>
<tbody>
<tr>
<td>150-frame “Suzie”</td>
<td></td>
<td></td>
</tr>
<tr>
<td>error-free</td>
<td>43.003</td>
<td>35.555</td>
</tr>
<tr>
<td>non-resilient error-prone</td>
<td>43.060</td>
<td>30.724</td>
</tr>
<tr>
<td>error-resilient with AIR</td>
<td>55.597</td>
<td>32.666</td>
</tr>
<tr>
<td>150-frame “Miss_America”</td>
<td></td>
<td></td>
</tr>
<tr>
<td>error-free</td>
<td>22.038</td>
<td>37.255</td>
</tr>
<tr>
<td>non-resilient error-prone</td>
<td>22.070</td>
<td>31.016</td>
</tr>
<tr>
<td>error-resilient with AIR</td>
<td>37.302</td>
<td>33.418</td>
</tr>
</tbody>
</table>

**Table 2.3** The detailed PSNR and bit rate variations for both test sequences

Figure 2.17 The subjective results of the 150th frames of “Miss_America” over a BER = 1e-03 random error channel, a- error-free; b- non-resilient error-prone and c- error-resilient with AIR

Figure 2.15 illustrates the results obtained from two sets of experiments which were conducted with the 150-frame “Suzie” and “Miss_America” sequences. The two graphs of the figure show three performance evaluations: error-free sequences as the reference plots, non-resilient error-prone and error-resilient with the use of AIR results. Each single plot was obtained by running the tests 10 times with 10 different random error pattern seeds at BER = 1e-03 to generate more reliable simulation conditions. Thus, the performance graphs show the average PSNR levels representing the 10 simulations. During the experiments, QP was set to 9 and 10 for “Suzie” and
“Miss_America”, respectively. Moreover, the pre-determined number intra (I) refresh MBs was 5 AIR MBs per frame for both of the video clips. With some additional functionality, these 5 AIR MBs were scanned across the whole video frame, so that the significant and high motion, but different regions of the picture, were refreshed in each frame. The simulations were run at a frame rate of 25 fr/s, with QCIF (176x144 pixels) size and in I-P-P-P-P-... format.

Table 2.3 demonstrates the objective PSNR and bit rate variations with and without the resilience for both of the sequences in detail. The corresponding subjective results, which comprise the last frames of the two video clips, can be seen in Figures 2.16 and 2.17.

2.6.3.2 Analysis of the Results

Figures 2.15.a and 2.15.b show the objective results of PSNRs against frame numbers for “Suzie” and “Miss_America”, respectively. These figures demonstrate the continuous quality degradations of the non-resilient video transmissions over a random BER = 1e-03 channel. On the contrary, the resilient schemes present better quality levels for both test sequences. The quality improvements provided by the use of AIR has been demonstrated to be around 2 dB and 2.5 dB on average in the error-prone “Suzie” and “Miss_America” transmissions, respectively. These numeric figures can also be seen in Table 2.3. The reason why “Miss_America” achieved a slightly higher quality level is that “Suzie” is a more active video sequence, in terms of motion, compared to “Miss_America”. Consequently, error destructions on video quality are more perceptible when there is higher motion in comparative tests as MVs are more sensitive to error occurrences. Moreover, predictive and motion compensated encoding and decoding render the entire sequence susceptible to error accumulations, as previously discussed in this chapter. Therefore, it can be noticed in both test results that towards the ends of the 150-frame sequences, the quality difference between the non-resilient and error-resilient streams become more distinguishable. The corresponding last frame extracts also confirm this particular difference in Figures 2.16-17.

Finally, Table 2.3 also demonstrate the output transmission bit rate increases due to the additional AIR MBs. As perceived, the AIR resilience algorithm increases the overall bit rate as it replaces a certain number of inter (P) MBs with the intra (I) refresh MBs. In this case, the number of refresh MBs was 5. Consequently, the output rates have been observed to increase by 13 kbit/s and 15 kbit/s on average for “Suzie” and “Miss_America”, respectively. However, this much rate increase is quite acceptable on the whole when the rate growth of the full I-frame encoding is considered.

Furthermore, Table 2.3 shows that output rates also slightly increased for the non-resilient error-prone transmissions when compared to the error-free ones. This was not incurred by the resilience algorithm as both transmissions were non-resilient. This is due to the fact that for simulation
purposes, the random error patterns were indeed applied to the VLC indices of the texture (DCT) and motion (MV) data rather than directly to the bitstream in order to maintain the synchronisation. In this way, the header and administrative data were protected. However, the corrupted VLC indices were often translated into longer codewords that eventually increased the overall output bit rate. This is due to the fact that commonly occurring events are normally encoded using short codewords so as to maintain a low bit rate. However, the newly corrupted indices refer to infrequently occurring events and hence, need to be represented by longer codewords. Nevertheless, as also noted from the table, the particular increases in the bit rates due to this fact are relatively insignificant amounting to the maximum levels of 0.06 kbit/s and 0.03 kbit/s for “Suzie” and “Miss_America”, respectively.

2.7 Concluding Remarks

The importance of using low bit rate video communications has been highlighted in this chapter. The essential background information has been given in relation to the user and network perspectives in low bit rate video communications. Moreover, some of the important applications are summarised in bullet-points. Two major low bit rate video coding standards, namely H.263 and MPEG-4, have been introduced along with their two motion-based optional modes of operations. Tests have been carried out to investigate the performance variations of these two optional modes, namely the advanced prediction mode and the unrestricted MV mode. It has been demonstrated that the video quality increases with the use of the two particular options, especially for high motion regions. In addition, the tests have also showed that MPEG-4 video coder performed slightly better compared to the H.263 video coder at similar bit rates.

This chapter has also discussed the error effects in video communications and the error-resilient video coding operation to provide robust transmissions. In this particular part, the emphasis has been put on two resilience algorithms: AIR and FCS, which will further be exploited in the following chapters of this thesis. Resilient video transmission tests have been carried out which demonstrated better quality levels compared to the non-resilient video transmissions. However, the improved QoS has been achieved at the expense of additional redundancy in the tested video streams which in turn results in an increased overall transmission bit rate.

Finally, a brief introduction has also been given on the source material used for the performance evaluations of the proposed algorithms throughout the thesis. The discussed topics in this chapter are believed to provide the reader with an understanding of the background concepts and a clear view of the further contents of this thesis.
Chapter 3

3 Homogeneous Video Transcoding

3.1 Introduction

Video transcoding algorithms that are particularly designed for VoD applications have been exhaustively presented in the literature for the past few years [Kan, Keesm, Morri]. These video transcoding algorithms have made use of the MPEG-2 digital video broadcasting (DVB) standard, bearing in mind that bit rate conversion of the previously encoded and stored VoD data might be needed for certain end-users. For the purposes of aiding the multimedia traffic planning, these video transcoding algorithms act as bit rate regulators in order that the congested or bandwidth-limited network(s) can accommodate the pre-encoded video streams [Assun1, Assun2, Fu, Lee2].

The referred planning problem of the multimedia traffic for congested nodes or bandwidth-limited networks will also exist in the future as it does now since the number and diversity of users and their needs vary in a wide range. However, the scope of the existing problem has very much proceeded lately from VoD and broadcast applications towards two-way mobile video communications. 3G mobile communication network infrastructures are promising to give support to video transmissions as well as voice and data communications. Thus, video transcoding algorithms have also become of importance as they provide the necessary transmission rate conversions between two or more communicating parties. A video transcoder of this kind allows the users of mobile communications to set up a connection regardless of their underlying network and connection statistics. Transmission rate adaptation is not performed at the encoder and decoder ends but during the transcoding process at a centralised unit within the network. Thus, this enables the service providers to employ much less complex encoder/decoder sets as opposed to the complex layered scalable video transmission equipment [Ghanb, Walke]. The latter produces multiple copies of the same video stream at different rates. On the contrary, transcoding provides similar solutions at the video gateway stage without incurring additional complexity at the source and user terminals.

Thus, in the light of these facts, this chapter discusses a video transcoding algorithm which provides a reasonable QoS level to each of the diverse bandwidth-constrained networks. The video transcoder algorithm designed in this chapter supports MPEG-4 video transmission as
MPEG-4 is believed to be the ubiquitous video compression standard for the next generation of mobile-wireless communications.

Chapter 3 is organised as follows: The second section discusses the video transcoding necessity in a heterogeneous type of entire networking scenario. In addition, this section also presents an overview of video transcoding and the types of video transcoding algorithms which have been reported in the literature so far. The third section gives an overview of the first category of the video transcoding techniques: homogeneous video transcoding. Moreover, this section discusses the different types of the homogeneous video transcoding algorithm. The fourth, fifth and the sixth sections look further into the bit rate, frame rate and resolution reduction/conversion schemes along with their advantages and disadvantages, respectively. The seventh section presents a multi-transcoder design for multimedia traffic planning purposes with the use of efficient homogeneous video transcoding schemes. The eighth section demonstrates the computer simulations and their results. Furthermore, this section also presents the discussions of these results obtained from numerous experiments carried out referring to the various parts of the chapter. Finally, the ninth section draws the concluding remarks.

### 3.2 Video Transcoding Necessity

The growth in the interest in multimedia communications and the low bit rate video support from the standardisation bodies and the manufacturers resulted in an incredible expansion in the number of multimedia applications. Moreover, these many applications induced an enormous diversity in the multimedia networking infrastructures. However, the diversity of the vast number of multimedia systems and the supporting various multimedia standards required a common platform to interact with each other. Consequently, an ITU standard for multimedia networking, namely H.323 [H323], has recently been established to provide this common platform for the necessary hand-shaking and compatibility between different networks and standards. This platform has been defined to be a multipoint control unit (MCU) which will bear necessary intermediate interoperability protocols at a centralised networking position. Video transcoding is one of the functionalities of the MCU to resolve certain network-related video communication problems. Video transcoding embraces the optimal solution of linking diverse multimedia networks to each other. The different networks on either side of the video transcoder might be operating with different compression standards or might have different characteristics. It is the video transcoder's ultimate aim to transfer an incoming video bitstream to the outgoing link without the need to fully decode and re-encode it. This operation achieves a low complexity, low delay and high efficiency interconnection of the multimedia networks of the same, similar or
diverse types. Thus, an interoperability solution can be accomplished between various networks or standards in a seamless way which is fully transparent to the end-users.

### 3.2.1 Video Transcoding Overview

Video transcoding is a method which makes the interoperability of different multimedia networks possible. It consists of operations which are capable of converting the format, resolution and/or the transmission rate of a compressed video sequence to various other formats, resolutions and/or rates. Therefore, the device which employs such an algorithm is called a video transcoder.

![Video Transcoder at the video gateway](image)

**Figure 3.1** A multimedia networking scenario using a video transcoder at the video gateway

The scaleable video encoding techniques described in [Ghanb, Radha] were the originating idea behind video transcoding. These schemes comprised a layered video encoder structure that provided different layers of compressed video streams at various bit rates. Thus, it was possible to produce a number of video streams compressed at different bit rates and which provide different levels of quality. At the time, this was necessary due to the wide deployment of VoD applications, where subscribers using bandwidth-restricted or congested networks, or even end-user problematic devices required access to one high resolution, high quality compressed video stream. In such cases, the most appropriate low bit rate stream was chosen at the expense of low resolution and quality. Layering consists of one base layer compressed with the lowest bit rate and quality level to provide the minimum requirements for a successful decompression of the video sequence. Several quality enhancement layers can be added to the base layer resulting in higher
bit rates. Depending on the varying network conditions, suitable bit rates were made up of the base layer plus a number of additional enhancement layers.

Apart from the complexity issue, the frequent changes in the network conditions and constraints, such as the congestion characteristics, forced the necessary adaptations to these changes to take place dynamically at a centralised point within the network. This specific location is referred to as a video gateway, as depicted in Figure 3.1. Such a device enables faster network responses whilst maintaining the user video encoders and decoders free of unnecessary complexities normally incurred by the scaleability features. A video gateway can consist of a single or a group of video transcoders operating together.

Figure 3.2 Video transcoding

The objective of video transcoding consists of changing the format, size, transmission rate and/or syntax of an incoming compressed video stream without fully decoding and re-encoding the video information. This idea is demonstrated in Figure 3.2. As a result, the complexity, processing power and delay incurred by this process are minimised whilst achieving improved QoS levels [Bjork, Kan, Keesm].

3.2.2 Video Transcoding Categories

In the past few years, [Assun3, Bjork, Dogan2, Kan, Keesm, Morri, Reyes1, Shan1, Warab, Youn1] proposed three major types of video transcoding algorithms. Out of these three, homogeneous video transcoding is by far the one which is mostly researched. A video sequence can be encoded at some high quality, high resolution and high bit rate at the service provider's source end, such as in VoD applications. However, a few or more network links might not be able to support any of the high performance bitstreams due to congestion reasons or network bandwidth constraints. Therefore, an interface between the two ends is needed for an achievable throughput. This interface is the homogeneous video transcoder which elaborates bit rate, frame rate, and/or resolution reductions according to the varying transmission conditions. Thus, the high performance bitstream can be relayed to the congested or constrained links by matching the input and output network characteristics.

Few papers in the literature present various methods of real-time homogeneous video transcoding for reducing the incoming bit rate of a compressed video bitstream. These papers also study the
performances of those real-time transcoders whilst giving experimental results. Real-time compressed video bitstream transcoding schemes are the partial fulfillment of the two different research projects: terrestrial trunked radio system (TETRA) project [Yeado] and advanced communication technology and services (ACTS) advanced television at low bit rates and networked transmission over integrated communication systems (ATLANTIC) project [Tudor], for supporting video in heterogeneous networks. Furthermore, an individual real-time video transcoder implementation research has also been presented in [Gopal].

Heterogeneous video transcoding, which is the second type of algorithm, has recently gained popularity due to the fast growing diversity in multimedia network structures and video coding algorithms. The heterogeneous video transcoder provides standard conversions. This is an exceptionally important issue as future global interoperability between asymmetric network topologies and multimedia communication standards will be supported using such a method. Therefore, the heterogeneity of the receivers will play an important role in the wide deployment of heterogeneous transcoders.

Lastly, the third type is also receiving increasing attention for deployment in error resilience applications for robust video communications: error-resilient video transcoding techniques. This requirement is induced by the demand for reliable multimedia communications over any platform, such as PSTNs, ISDNs, Internet or particularly the highly error-prone mobile-wireless environments. Thus, support for reliable video communications also need the video transcoders to be tailored for error resilience purposes.

### 3.3 Homogeneous Video Transcoding

The objective of homogeneous video transcoding schemes is to further compress a pre-encoded video stream by reducing its bit rate, frame rate and/or resolution. This rate reduction is achieved whilst preserving the syntax format and compression characteristics with which the incoming video stream was initially encoded. For this reason, such rate reduction tools are known as homogeneous transcoding methods. This technique is illustrated in Figure 3.3.

Homogeneous video transcoding is applied to reduce the size of an encoded incoming bitstream. In this way, a video encoder is de-coupled from the transmission network constraints, such as congestion, packet losses and varying user demands over one type of network [Assun1]. However, the approach is also naturally applied to the present heterogeneous communication networks, such as circuit-switched, asynchronous transfer mode (ATM), mobile and Internet, as the forthcoming multimedia telecommunication services are expected to use a great deal of video material in the
compressed format for storage and transmission. Thus, a very wide range of video services from very low bit rate (under 64 kbit/s) to high definition television (HDTV) systems with a bit rate of 20 Mbit/s require matching source and network characteristics. Consequently, homogeneous video transcoders are employed to achieve necessary rate conversions with low delay and low complexity whilst maintaining reasonable quality levels. In this manner, traffic planning for congestion control at network nodes and rate controlling can be yielded. Moreover, transcoding helps the communication service providers prevent the packet losses due to the network congestion by providing the necessary bit or frame rate reductions.

Figure 3.3 Homogeneous video transcoding

A typical scenario where homogeneous video transcoding can be efficiently employed is the multipoint video conferencing. In this case, a high quality, high bit rate video bitstream is re-encoded by the transcoder in order to transmit the video data at various lower rates, through communication channels of different capacities. Another technique for combining multiple video bitstreams for the purpose of video conferencing is called a coded domain combiner. This is a low complexity technique which simply concatenates the incoming video streams. Thus, the resultant bit rate consists of the combined bit rates of each incoming video stream. However, in the reverse direction, this technique allocates an equal bandwidth to each of the users taking part in the video conference, regardless of their individual requirements and/or available bandwidth. In contrast, video transcoding partially decodes each of the incoming video bitstreams, combines them and re-encodes all the information as a single video stream. Therefore, this method provides an efficient use of the available bandwidth of each user. During the re-encoding process, an active conference participant is allocated a higher number of bits. As a result, every user experiences a uniform video quality. However, the video transcoded method incurs a higher complexity when compared to the coded domain combiner, as presented in [Sun1].

Likewise, [Lin] proposes a dynamic rate control scheme for video transcoding employed in multipoint video conferencing. This method aims to enhance the visual quality of the participants and other regions of interest by identifying the high motion areas, such as the active conference
participants, from the multiple incoming video streams. These regions of interest are then transcoded with a more optimised bit allocation approach at the expense of relatively reduced qualities for the inactive or less active users.

Homogeneous video transcoding methods are exploited for a successful provision of a compromise between the user requirements and the capacities of the diverse networks, as discussed in this section. This compromise is accomplished through various types of homogeneous transcoding techniques which will be discussed in the forthcoming sections in detail.

The recent progress and advance in the research of homogeneous video transcoding schemes have been mainly driven by the increasing popularity of VoD applications. This is due to the fact that the VoD data is encoded with high quality, high resolution and high bit rate (i.e. a few Mbit/s) using the MPEG-2 standard. However, in some cases, such as in bandwidth-limited networks, during congestion periods or when the recipient is not able to support such a high rate, transmission rate reduction becomes a necessity. Furthermore, in the case when the user device is not equipped with the right display, the size of the incoming video data needs to be reduced to a smaller size (i.e. CIF to QCIF).

Thus, different types of homogeneous video transcoding algorithms arise as to match the source-driven transmissions to the requirements of the end-user equipment. There are three major types of homogeneous video transcoding algorithms, namely bit rate, frame rate and resolution reduction techniques.

### 3.4 Bit Rate Reduction

Among the different types of video transcoding schemes, bit rate reduction algorithms have so far been the only intensely researched schemes. This is due to a growing popularity in VoD applications, which are encoded at a relatively high rate, as described in the previous section. Various examples of standard rate conversions for high bit rate video transmissions have been presented in the literature. These include conversions from a few Mbit/s to a few hundred kbit/s. However, due to the wide-spread deployment of the mobile-wireless and satellite communication systems, algorithms which convert high bit rates to low rates and low bit rates to even lower rates (i.e. from a few hundred kbit/s to a few ten kbit/s) have also gained the deserved attention.

A number of ways in which the incoming bit rates are converted to lower rates have been investigated. [Assun4] describes how the DCT coefficients with the highest frequencies are arbitrarily selected and then simply discarded by truncating. On the other hand, [Assun4, Nakaj,
Sunl, Werne] perform the down-scaling by re-quantising the transform coefficients with a coarser quantisation step-size. Both of these techniques achieve a reduction in the number of the DCT coefficients by forcing a selected few to zero. This results in fewer number of bits from the transcoder output.

The re-quantisation process of the transform coefficients, which was briefly discussed in the previous paragraph, employs the same scalar MPEG quantisers of the MPEG video standard. A similar re-quantisation process has been introduced by [Lois]. However, this approach makes use of a lattice vector quantiser (LVQ) in order to extend the MPEG compression capabilities whilst providing an acceptable quality. LVQ is a multi-dimensional generalisation of uniform step scalar quantisers which performs particularly well for uniformly distributed data, resulting in minimal output distortion. Since the quantisation errors are uniformly distributed in the transcoded pictures, LVQ produces less visible artefacts at the edges. In addition, the codebook storage is not required and the search complexity is simplified. However, the output bitstream resulting from the LVQ is MPEG incompatible. In order to correct for this incompatibility, a low complexity, low cost user interface is required. This interface involves the LVQ decoder and the MPEG entropy encoding engine. The resulting output bitstream can then be fed into an MPEG video decoder directly at the end of the telecommunication system.

The following sub-sections describe the five different techniques that are currently employed for bit rate reduction. The first sub-section discusses the conventional cascaded fully decoding/re-encoding scheme. The last four schemes are low complexity straightforward transcoding methods which are investigated for fixed quality and hence, variable rate conditions.

### 3.4.1 Cascaded Fully Decoding/Re-Encoding Scheme

The conventional tandem operation of two video networks is depicted in Figure 3.4. The whole process comprises two separate operations on the incoming compressed video stream, which involve a complete decoding process cascaded with a fully re-encoding process. Prior to the re-encoding stage, the scheme performs the required re-ordering and re-sizing of the bitstream. Thus, this non-optimised scheme performs a re-evaluation of both the video frame headers and the MB headers.

This type of cascaded operation involves:

- Complex frame re-ordering which requires a complete constitution of the video frame parameters from raw data.
- Full-scale (±16 pixels) motion re-estimation.

This results in a scheme which:
Chapter 3 Homogeneous Video Transcoding

- Has the highest complexity compared to the other four schemes to be introduced.
- Incurs long processing time and high power consumption, causing significant delay and low quality pictures due to the fact that the motion re-estimation is performed using the reduced quality decoded pictures.

![Figure 3.4 Cascaded fully decoding/re-encoding scheme versus transcoding](image)

Figure 3.4 also displays the difference between the cascaded decoding/re-encoding method and the transcoding scheme, where the decoder and the re-encoder blocks are merged into a lower complexity approach.

### 3.4.2 Transcoding with Re-Quantisation Scheme

The second bit rate reduction scheme which is described in this sub-section is the transcoding method which employs direct re-quantisation. This technique is also referred to as the open-loop transcoding algorithm because it involves straightforward transcoding without any feedback, as shown in Figure 3.5. In contrast with the first scheme, this algorithm partially decodes the input video stream already compressed. As a matter of fact, only the variable length Huffman codewords, which represent the DCT coefficients of the video data, are decoded. Once this is achieved, the decoded transform coefficients are first inverse zigzag scanned and then inverse quantised using the quantisation parameter with which the original video data was initially quantised. The de-quantised coefficients are then re-quantised with a different quantiser and zigzag re-scanned. This new quantiser has a reduced number of levels resulting in a coarser quantisation with a reduced output rate. Prior to transmission, the re-quantised coefficients are Huffman re-encoded. The transcoding algorithm does not require:

- Complex frame re-ordering.
- Full-scale (±16 pixels) motion re-estimation.

Therefore, open-loop transcoding incurs:

- The lowest complexity compared with all the schemes described in this section, due to its simple transcoding mechanism.
- Low delay and power consumption resulting from the small processing time required for this algorithm.
In this transcoding scheme, the original MVs and the video frame headers are left unmodified. On the other hand, due to the fact that some of the originally encoded MBs may be skipped during the coarser re-quantisation process, the MB headers need to be re-calculated. The following criteria are used to select the MB types during the re-evaluation process:

- an MB which was originally skipped is also skipped during the transcoding,
- an intra (I) MB is transcoded as an intra MB, and
- an inter (P) MB is however transcoded as inter (predictive) or intra MB depending on the motion data and residual transform coefficients. If this information is not present, the inter (P) MB is skipped.

One of the major advantages of open-loop transcoding is the fact that it operates in the coded domain. For this reason, it is possible to implement a simple, fast operating transcoder with a lower complexity. However, the simplicity of this algorithm is counterbalanced by the reduced level of QoS, incurred by the picture drift effect during prediction. The drift effect results from the use of different quantisers at the encoder and the transcoder which produces a mismatch in the reconstructed pictures [Assun4, Sun2]. The degradation in the quality of the transcoded video data needs to be minimised in order to provide an improved QoS. In order to demonstrate how this is accomplished, the following sub-section includes the mathematical derivation of the drift problem. This theory is used to achieve drift-free transcoding schemes, which are also briefly discussed in the next sub-section.

### 3.4.2.1 The Drift Effect

The drift problem in video transcoding has been intensely investigated [Assun4, Bjork, Sun2]. It is a result of the distortion incurred in the reconstructed pictures following the transcoding operation. This distortion results from the use of two different quantisers and accumulates with the number of frames processed. Referring back to Figure 3.5, the DCT coefficients are first de-quantised using the original quantiser. These coefficients are then re-quantised using a smaller set of coarser quantisation levels in order to achieve a bit rate reduction. This operation causes a deterioration in the output quality of the transcoder. However, this degradation effect is completely different from the quality reduction resulting from one decoding/re-encoding operation.
performed during transcoding. Such a process inevitably introduces some loss in the output quality since the re-encoding operation is performed on the reconstructed video data which is inherently of lower quality. The reduction in quality is due to the quantisation process which is a lossy operation. The distortion in the de-quantised data is a well-predicted occurrence which should clearly be distinguishable from the picture drift effect.

On the contrary, the drift effect is a prediction-oriented problem which is only incurred by the transcoding operation of the predictive (P) frames. Therefore, it is important to note that the drift effect only occurs in open-loop transcoding. The degradation effects accumulate within the video stream due to the fact that the current standardised video codecs, such as H.261/263/263+/263++ and MPEG-1/2/4, employ predictive coding. The transcoding of intra (I) and bi-directional (B) frames do not contribute to the drift problem. This is due to the fact that I-frames are not prediction-dependent but are encoded with reference to themselves. In addition, B-frames are not used in the prediction of future frames either.

A simple approach which counteracts the occurrence of the drift involves a regular and frequent transmission of intra frames, which aim to refresh the transcoder memory. However, this solution increases the amount of data in the video stream which inevitably increases the bit rate and hence, defies the objectives of bit rate reduction and video transcoding.

A more efficient way of tackling the drift problem is to design a video transcoding algorithm which by means of a feedback loop, it self-eliminates the drift effect. This approach, which has been extensively researched [Assun3, Assun4, Bjork, Youn1, Youn2, Youn3], is discussed in the following sub-section following the mathematical analysis of the drift effect.

[Assun4] analyses the drift problem and states that the end-decoder is the same as the local decoder situated in the encoder. Therefore, one can conclude that in an error-free environment, the reconstructed pictures at the decoder are the same as the ones produced by the local decoder at the encoder. This can be mathematically represented as:

\[ R_{P_n}^d = R_{P_n}^e, \quad n = 0, 1, \ldots, N - 1 \]  

(3.1)

where \( R_{P_n}^d \) and \( R_{P_n}^e \) represent the reconstructed pictures at the decoder and encoder, respectively. \( N \) is the total number of processed video frames. The reconstructed picture for a predictive (P) frame can also be represented by some prediction error, \( e_n \), together with a motion-compensated prediction term, \( \text{MCpred} \), as follows:

\[ R_{P_n}^d = R_{P_n}^e + \text{MCpred}(R_{P_{n-1}}^d), \quad 1 \leq n \leq N - 1 \]  

(3.2)

However, since an intra (I) frame is encoded with respect to itself without any motion compensation and prediction, the reconstructed picture of an I-frame is represented by:
When bit rate reduction is accomplished with the use of an open-loop transcoding algorithm, the above equations need to be modified so as to take into consideration the transcoding distortion. Therefore, the reconstructed images at the end-decoder and at the local decoder at the encoder side are no longer the same. Instead, the following derivations apply:

\[ R_{p}^{d}_{n} = R_{p}^{e}, \quad n = 0 \] (3.3)

\[ R_{p}^{d}_{n,\text{distorted}} \neq R_{p}^{e}, \quad n = 0 \]

\[ R_{p}^{d}_{0,\text{distorted}} = R_{p}^{e} + t_{0}^{\text{distort}}, \quad 1^{st} I_{frame} \]

\[ R_{p}^{d}_{1,\text{distorted}} = e_{1} + t_{1}^{\text{distort}} + \text{MCpred}(R_{p}^{d}_{0,\text{distorted}}), \quad 2^{nd} P_{frame} \] (3.4)

\[ R_{p}^{d}_{1,\text{distorted}} = e_{1} + t_{1}^{\text{distort}} + \text{MCpred}(R_{p}^{e} + t_{0}^{\text{distort}}) \]

\[ R_{p}^{d}_{1,\text{distorted}} = e_{1} + t_{1}^{\text{distort}} + \text{MCpred}(R_{p}^{e}) + \text{MCpred}(t_{0}^{\text{distort}}) \]

where \( \text{MCpred} \) is assumed to be a linear operation. The first two derivations in Equation (3.4) demonstrate that the transcoding distortion, \( t^{\text{distort}} \), consists of the difference between the current picture at the end-decoder and the one at the local decoder of the encoder. The remaining derivations show that for the subsequent predictively encoded picture (P), the reconstructed picture at the decoder does not only consist of the motion-compensated previous I-frame combined with the prediction error. It also includes the transcoding distortion of the current frame and the motion-compensated predictive version of the previous frame. The latter distortion term is referred to as the residual of the transcoding distortion resulting from the previous frame and is represented as:

\[ \Delta_{1} = \text{MCpred}(t_{0}^{\text{distort}}) \] (3.5)

where \( \Delta \) is the drift error in the picture. Similarly, the drift error for the second frame can be defined as:

\[ R_{p}^{d}_{2,\text{distorted}} = e_{2} + t_{2}^{\text{distort}} + \text{MCpred}(R_{p}^{d}_{1,\text{distorted}}), \quad 3^{rd} P_{frame} \]

\[ \Delta_{2} = \text{MCpred}(t_{1}^{\text{distort}} + \text{MCpred}(t_{0}^{\text{distort}})) \] (3.6)

Therefore, from Equation (3.6), one can conclude that the drift error consists of the cumulative behaviour within a predictive video stream and it can generically be applied for any picture as follows:

\[ \Delta_{n} = \text{MCpred}(t_{n-1}^{\text{distort}} + \text{MCpred}(t_{n-2}^{\text{distort}} + ... + \text{MCpred}(t_{0}^{\text{distort}}))) \] (3.7)
3.4.2.2 Drift-Free Transcoder Algorithm

The study and analysis of the drift problem have led to a simple video transcoding design which operates drift-free. [Assum4] presents a solution where the drift error is corrected with the use of a feedback loop. The block diagram of the proposed drift-free algorithm, which is depicted in Figure 3.6, consists of a very basic configuration. It comprises two separate stages, namely a decoding block at the input and a re-encoding block at the output of the homogeneous video transcoder. However, both of these blocks are not the same decoder and encoder blocks that were employed in the cascaded fully decoding/re-encoding scheme. Instead, they perform partial decoding and encoding operations. The feedback loop which corrects the drift error is applied within the re-encoding block.

![Block diagram of drift-free transcoder](image)

Figure 3.6 A basic drift-free homogeneous video transcoder block diagram

Referring to Figure 3.6, the input video stream encoded at rate $R_1$ is decoded in the first stage and is then re-encoded in the second stage with a coarser quantisation $Q_2$, achieving a reduced output rate $R_2$. Therefore,

$$Q_2 > Q_1 \Rightarrow R_2 < R_1$$

(3.8)

Moreover, $R_1'$ and $R_2'$, which represent the bit rates after the inverse quantisation processes at points 1 and 2, respectively, can be defined as:

$$\Delta A_n = A_n - A_{n-1}$$

$$R_1' = DCT(\Delta A_n)$$

(3.9)
where $\Delta A_n$ is the reconstruction error from the incoming bitstream and $DCT$ is the transform operator. The terms $A_n$, $A_{n-1}$, $B_n$, and $B_{n-1}$ represent the locally decoded current and previous pictures of the input and the output streams, respectively. Similar definitions can also be given for $R_i'$ as follows:

$$
R'_2 = DCT(A_n - B_{n-1})
$$

$$
R'_2 = DCT(\Delta A_n + A_{n-1} - B_{n-1})
$$

$$
R'_2 = DCT(\Delta A_n) + DCT(A_{n-1} - B_{n-1})
$$

$$
R'_2 = R'_1 + DCT(A_{n-1} - B_{n-1})
$$

where the transform operation is considered to be a linear operation. The last definition of Equation (3.10) is particularly significant because it indicates that the incoming rate can directly be used without the need for the whole decoding process. A block diagram of the simplified transcoder structure, which exploits this feature, is shown in Figure 3.7. Since the two loops are symmetrical, they can be merged to further simplify the transcoding process. The second loop shown in the figure, is the feedback loop and for this reason, this kind of a transcoding scheme is referred to as a closed-loop video transcoding algorithm. The purpose of this loop is to accumulate the drift errors introduced by the different quantisation factors and add them back into the next frames following the motion compensation. This feedback compensates for the picture drift and hence, stops its accumulative effects throughout the video sequence.

![Figure 3.7 Simplified drift-free video transcoder block diagram](image)

In comparison with the conventional cascaded fully decoding/re-encoding scheme, this structure provides a simplified drift-free video transcoding algorithm with a reduced complexity. However,
the additional DCT/Inverse DCT (IDCT) operations required in this scheme tend to increase the complexity of the transcoder. Furthermore, additional storage in the form of a frame buffer is required for the locally reconstructed video frames. Due to the fact that the reconstruction is performed in the pixel domain, the DCT/IDCT operations are inevitable. Nevertheless, [Achar, Assun5, Assun6] propose a few drift-free video transcoding algorithms which operate in the DCT domain and which do not require DCT/IDCT operations. However, these schemes have not been reported to offer a less complex solution either [Björk, Sendal].

3.4.3 Transcoding with Motion Data Re-Use Scheme

The motion data re-use scheme is one of the three drift-free algorithms employed for video transcoding. This scheme uses the original quantiser to partially decode the incoming compressed video information up to the motion data, as described in the previous sub-section. The decoded DCT coefficients are then re-quantised with different quantisation levels in order to yield the required output bit rate reduction. Prior to transmission, the re-quantised DCT coefficients are re-encoded with Huffman coding methods.

So far, the description of this particular transcoding algorithm closely resembles the previously discussed open-loop transcoding method. However, in contrast with the formerly described scheme, the motion data re-use method employs an additional feedback loop. The aim of this loop consists of resolving the accumulated mismatch errors between the reconstructed images of the source and those of the transcoder incurred by the use of different quantiser levels. The correction of the drift error is performed in the same way as discussed in the former sub-section. Figure 3.8 depicts the structure of the overall system. The feedback loop consists only of the frame buffer block which contains the previously reconstructed video frames. These video frames are reconstructed with the new quantiser levels set by the video transcoder itself in order to achieve the required amount of bit rate reduction. Therefore, compared to the open-loop transcoding method, this scheme needs to store the previously reconstructed frames in the pixel domain. Therefore, closed-loop algorithms consist of both pixel domain and coded DCT domain operations. Consequently, even though the transcoding with motion data re-use scheme does not involve:

- Complex frame re-ordering.
- Full-scale (±16 pixels) motion re-estimation.

operations, it incurs:
Higher complexity than the simple re-quantisation method, but much lower complexity than the cascaded fully decoding/re-encoding and also the forthcoming closed-loop schemes with motion re-handling.

- A low processing time and power consumption in addition to a small amount of delay.

The motion data re-use scheme is the least complex drift-free video transcoding algorithm due to the fact that it does not include any kind of motion data re-evaluation. The incoming original MVs are simply re-used without any modification. However, this results in a degraded transcoded video quality despite the employment of the drift correction loop. The reduction in quality is incurred by the deviation of the input original MVs from their optimal values due to the differential reconstruction errors [Bjork, Youn3]. This means that in some cases, the original MVs may not be optimal for the new set of quantiser levels as they may point to wrong blocks within a video frame. Therefore, the transcoding quality can further be improved by the refinement of the incoming MVs, as described in a forthcoming sub-section. Furthermore, MB headers are re-evaluated, as discussed earlier.

### 3.4.4 Transcoding with Motion Data Re-Estimation Scheme

Similar to the motion data re-use algorithm described in the previous sub-section, the motion data re-estimation scheme also comprises a closed-loop for drift error correction. However, this particular scheme, includes a full-scale estimation of the new MVs instead of using the original video motion data. The new motion estimation for the new conditions is performed for a full size MV search window, which is about ±16 pixels around the candidate block whose motion is being estimated. The motion estimation process was briefly explained in Chapter 2. One of the advantages of this scheme consists of the fact that it does not involve:

- Complex frame re-ordering.

However, it performs:
Chapter 3 Homogeneous Video Transcoding

- Full-scale (±16 pixels) motion re-estimation.

For this reason, the motion data re-estimation method incurs:

- Much higher complexity than the simple re-quantisation and the motion data re-use methods, but lower complexity than the cascaded fully decoding/re-encoding scheme.
- A considerable amount of processing time and power consumption with a substantial amount of delay.

![Transcoding with motion data re-estimation scheme](image)

Moreover, as illustrated in Figure 3.9, the motion re-estimation block within the drift correction loop, reduces the degradation effects on the transcoding quality that occurred with the motion data re-use scheme due to the use of non-optimal MVs. Thus, this scheme is able to estimate and hence, select the most suitable MVs for the modified quantiser levels of the video transcoder. In addition, this scheme also re-evaluates MB headers due to the reasons explained earlier.

### 3.4.5 Transcoding with Motion Data Refinement Scheme

The two closed-loop schemes described so far reduce the drift effects on the transcoded video quality. However, the first scheme produces non-optimal MVs that degrade the output quality. On the other hand, the second one resolves the quality degradation problems with full-scale motion data re-estimation at the expense of added complexity. These two methods show that the sole use of a drift correction loop is often not sufficient for an improved QoS with low complexity. As a solution, [Bjork, Youn3, Youn4] propose a further MV refinement process to be integrated into the system. Since this process can only be accomplished in the pixel domain with the use of a locally reconstructed video frame, DCT domain transcoding algorithms fail to achieve motion data refinement [Senda].

The motion refinement algorithm is detailed in the following sub-section. This sub-section concentrates on the architecture of the video transcoder which exploits the useful features of this algorithm. Figure 3.10 depicts the motion data refinement block within the transcoding scheme. In
contrast with the previous scheme, this particular method re-uses the original MVs. Due to the fact
that the direct re-use of these non-optimal vectors causes a reduction in the video quality, these
vectors are refined. The refinement process is performed around the non-optimal MVs in order to
select the best representing finer vectors. The major advantages of transcoding with motion
refinement consist of the fact that this scheme does not require:

- Complex frame re-ordering.
- Full-scale (±16 pixels) motion re-estimation.

However, it performs:

- Small-scale (±1, 2, 3, 4, etc. pixels) motion refinement.

As a result, the scheme incurs:

- Higher complexity than the simple re-quantisation and the motion data re-use
  methods, but lower complexity than both the transcoding with full motion data re-
  estimation and the cascaded fully decoding/re-encoding schemes.
- Moderate processing time and power consumption with some delay.

![Figure 3.10 Transcoding with motion data refinement scheme](image)

The complexity and hence, the processing delay increase with the increasing motion refinement
window size, as demonstrated in the succeeding sub-sections. It is also important to note that the
MB headers require re-evaluation, as described in the previous schemes, in order to ensure that the
correct MB types are selected.

### 3.4.5.1 The MV Refinement Algorithm

This algorithm involves the refinement of the incoming MVs which were estimated and encoded
using the original quantiser. Due to the re-quantisation process, which employs a different
quantiser, these vectors might be non-optimal. As a result of the differences in the quantisation
steps, the initially estimated MVs might point to the wrong blocks or MBs within a video frame.
Therefore, a refinement process is required and is accomplished by means of a search window
limited to a small number of pixels which are pointed to by the original MVs.
The proposed solution makes use of the received original MVs and hence, avoids the added complexity of re-estimating the motion data for the new conditions. In addition, a small MV search window is sufficient to produce the necessary quality improvement whilst significantly reducing the complexity.

Figure 3.11 A complete video transcoding system

[Youn3] describes the need for MV refinement whilst detailing the theoretical and mathematical aspects of the problem. In this particular reference, the MV set \((I_a, I_p)\) for the encoder is obtained by:

\[
(I_a, I_p) = \{ \min_{(a,b) \in W} \text{SAD}_e(a,b) \}
\]

\[
\text{SAD}_e(a,b) = \sum_h \sum_v |P^e_c(h,v) - R^p_c(h+a,v+b)|
\]

where \(h\) and \(v\) are the horizontal and vertical variables in the motion estimation process. \(P^e_c(h,v)\) represents a pixel in the current frame while \(R^p_c(h+a,v+b)\) represents a pixel in the previously reconstructed reference frame, which is displaced by \((a, b)\). The superscripts \(c\) and \(p\) refer to the current and the previous frames, respectively. The subscript \(e\) indicates the encoder block whilst \(W\) represents the fixed search window range.

By converting the subscript \(e\) to \(t\), the transcoded MV set \((T_a, T_p)\) can be obtained by similar equations as follows:

\[
(T_a, T_p) = \{ \min_{(a,b) \in W} \text{SAD}_t(a,b) \}
\]

\[
\text{SAD}_t(a,b) = \sum_h \sum_v |P^e_t(h,v) - R^p_t(h+a,v+b)|
\]

64
From Figure 3.11, it is evident that the reconstructed picture within the transcoder, $R_e$, is an input to the re-encoding part of the transcoder block. Therefore, it is equivalent to the current picture, $P_t$, of the transcoder and:

$$SAD_t(a,b) = \sum_h\sum_v \left| P_t^e(h,v) - R_t^p(h+a,v+b) \right| + SAD_t(a,b) - SAD_t(a,b)$$

$$SAD_t(a,b) = \sum_h\sum_v \left[ P_t^e(h,v) - R_t^p(h+a,v+b) \right]$$

$$SAD_t(a,b) = \sum_h\sum_v \left[ P_t^e(h,v) - R_t^p(h+a,v+b) \right]$$

where, since $R_e = P_t$,

$$\Delta_e^e(h,v) = R_t^e(h,v) - P_t^e(h,v) \quad (3.13)$$

$$\Delta_t^p(h,v) = R_t^p(h,v) - P_t^p(h,v) \quad (3.14)$$

In the above equation, $\Delta_e^e(h,v)$ represents the quantisation error of the current frame in the encoder whilst $\Delta_t^p(h,v)$ identifies the quantisation error of the previous frame in the transcoder. Thus, the optimal MVs in the transcoder are correlated with the incoming non-optimal MVs and the quantisation errors incurred by the encoder and the transcoder.

Following these observations, the MVs produced by the transcoding operation can be formulated as follows:

$$SAD_t(T_x,T_y) = SAD_e(I_x,I_y) + \sum_h\sum_v \left| \Delta_e^e(h,v) - \Delta_t^p(h+x,v+y) \right|$$

$$SAD_t(T_x,T_y) = SAD_e(I_x,I_y) + SDQE$$

where SDQE represents the sum of the differential quantisation error. In Equation (3.14), this term was referred to as $\Delta$ functions. Equation (3.15) shows that the direct re-use of the incoming original MVs, $(I_x, I_y)$, results in non-optimised outgoing MVs, $(T_x, T_y)$, due to the existence of differential quantisation errors.

A small reduction in the bit rate results in minor output quality deterioration due to the fact that the differential quantisation error is also small. On the contrary, a large asymmetrical downscaling of the bit rate produces significant degradations in performance caused by the higher differential quantisation error. Thus, the reduction in quality is proportional to the differential quantisation error.
Figure 3.12 demonstrates a typical scenario where refinement can be employed. The dashed vector represents the non-optimal MV at the input of the transcoder. Refinement is achieved by means of a search around the pixel pointed to by the initial MV, which in the figure is represented by a grey dot. The search is carried out with a refinement window of ±3 pixels for this particular example illustrated in Figure 3.12. Motion refinement operations, such as SAD calculations and best matching block search, are performed within a defined search window. If a better matching block with the minimum SAD is found, this new pixel, which is represented by a block dot, is chosen instead of the non-optimal MVs as the reference for this newly estimated block position. Thus, a new MV set is estimated, as shown by the straight vector in the figure. The difference vector between the non-optimal one and the recently estimated one gives the refining MV, as shown by the vector in bold. The following basic vector operation, found in [Youn3], is used to obtain the refined MV

$$\vec{OB} = \vec{OA} + \vec{AB}$$

(3.16)

where $\vec{AB}$ represents the refining MV, as displayed in the figure.

### 3.4.5.2 The Effects of Refinement Window Size Variation on Transcoding Quality

A typically small MV search window size is selected in order to limit the complexity incurred by the operations in the video transcoder [Youn3, Youn4]. Such small window sizes provide reasonable performance levels for the refinement of incoming non-optimal MVs whilst maintaining a low complexity.

However, the use of a larger window size yields an improvement in performance. The amount of enhancement in quality becomes more significant for high motion activity scenes. This is due to the fact that a greater search range can provide a closer match of a candidate block or MB to a
previously reconstructed one at a different location. Moreover, transcoding quality improvement increases with the increasing refinement window size in the case that bit rate reduction asymmetry is significant. However, a bigger window size also increases the output bit rate which has already been reduced by the transcoder. This is due to the fact that a larger window contains a higher number of pixels. Therefore, a trade-off is required amongst the bit rate reduction, operational complexity and improved transcoding quality. The graphs in Figure 3.13 demonstrate the average PSNR levels and bit rates versus different refinement window sizes for the 200-frame “Foreman” sequence whilst the plots in Figure 3.14 present the similar variations for the 150-frame “Suzie” sequence. These two particular video test sequences, with high and moderate motion activities, were deliberately chosen to enhance the effects of the increasing refinement window size on the video transcoding quality.

![Figure 3.13](image)

**Figure 3.13** Average PSNR and bit rate variations of the transcoded “Foreman” sequence, originally encoded at an average of 116 kbit/s and 25 fr/s, versus various refinement window sizes

![Figure 3.14](image)

**Figure 3.14** Average PSNR and bit rate variations of the transcoded “Suzie” sequence, originally encoded at an average of 64 kbit/s and 25 fr/s, versus various refinement window sizes

The results displayed in Figures 3.13-14 show a slight improvement in the PSNRs for the increased refinement search ranges. However, this is achieved at the expense of a higher bit rate, as observed from both figures. The increase in the PSNRs is particularly noticeable when the reduction in bit rate is large, i.e. from very high rates to very low rates, such as from 116 kbit/s down to 47-43 kbit/s for “Foreman” and from 64 kbit/s down to 28-26 kbit/s for “Suzie”. On the other hand, when the difference between the input and output rates is not high, both the PSNR and
the bit rate do not experience significant changes with the increase in MV refinement search window size.

This very important and useful information about the behaviour of the transcoding quality versus varying refinement window size can further be exploited to give better rate transcoding performances. This will indeed be discussed in the following part.

3.4.5.3 Tuning of Video Transcoding Quality by Self-Adjusting Search Window Approach

Bit rate management can be achieved through the re-quantisation of the transform coefficients. In the case that the input video stream requires a reduction in overall size due to the bandwidth constraints, the video transcoder re-quantises the transform coefficients in a coarser way. Similarly, a finer re-quantisation results in an increase in the output bit rate for supporting networks with larger capacity.

Direct re-quantisation introduces a picture drift that continuously deteriorates the video quality, as demonstrated in a former sub-section. Thus, the use of a drift correction loop with the MV re-use is a general acceptance. Moreover, non-optimal MVs are also refined for a better match between the reconstructed images of the source and the transcoder. The conventional MV refinement comprises pre-determined fixed-size small search windows (±1,2,3,4 pixels) around the blocks of a video frame to which the incoming non-optimal MVs point.

In view of these facts, a proposed novel technique of self-adjusting search window approach can now be further detailed. Numerous test results obtained from various standard video sequences proved that motion refinement gave better performances as the MV search window size increased. A collection of these test results have already been presented and discussed in the preceding part. This event particularly took place during a high asymmetrical degree of bit rate reductions. It happened due to a better match of blocks in a wider search range in comparison with their counterparts in the previous frames. However, the wider the search range, the higher the bit rate. Thus, the refining window should not be set free to enlarge up to a full MB size (±16 pixels) to avoid the increase in the bit rate and contravene the main purpose of transcoding whilst keeping the complexity minimal.

High motion areas cause the blocks of a frame to move more pixels than a fixed-size small search window can trace. Unlike in conventional techniques, the MV search window size is set flexible in this approach for finer refinements. The MV refinement window adapts and re-sizes itself to the amount of motion data that the transcoder is processing for a particular video frame. This adaptation is achieved via the computation of the average translational motion difference per frame, in terms of pixels, between the incoming non-optimal MVs and the refined MVs within a
certain search window size. In other terms, average motion difference is a measure of the mobility of the blocks in a frame. This computed average is then fed back to the system, as shown in Figure 3.15, which controls the adjustment of the refining search window area. Thus, a larger average motion difference enlarges the window size by 1 pixel in both directions while a smaller one reduces it by the same amount. Naturally, zero difference results in maintaining the previous size. Separate handling of the vertical and horizontal motions did not give any better performance than the combined motion during the tests as a block would also be expected to make a diagonal move.

![Figure 3.15 Block diagram of a video transcoder with refining window adjuster](image)

Test results were obtained for different standard video sequences encoded at 25 fps and various bit rates using the advanced prediction and unrestricted modes of the MVs. The video transcoder was designed to work on the MPEG-4 standard with the self-adjusting refinement window size confined to a lower limit of ±1 pixel, and an upper limit of ±8 pixels (a block size or half an MB size) in both directions. The selection of the upper limit boundaries was determined during the tests by the optimal compromise amongst the quality, complexity and the bit rate.

Tables 3.1-3 display the simulation results of the 150-frame “Suzie” I/II and the 200-frame “Foreman” I/II sequences indicating the transcoding operations from 64 to 33/25 kbit/s and from 116 to 68/42 kbit/s on average, respectively. The lowest and the highest fixed window boundary test results are also presented for comparisons to the proposed self-adjusting scheme results. The test results for the other fixed window sizes between ±1 and ±8 pixels are not shown as they displayed monotonously increasing values with enlarging sizes for all the three tables. This has already been demonstrated within the former part where the resulting effects of the varying search window size on the transcoding quality and the bit rate were sought.

<table>
<thead>
<tr>
<th>PSNR [dB]</th>
<th>“Suzie” I</th>
<th>“Suzie” II</th>
<th>“Foreman” I</th>
<th>“Foreman” II</th>
</tr>
</thead>
<tbody>
<tr>
<td>fixed ±1 pixel</td>
<td>33.938</td>
<td>31.841</td>
<td>31.947</td>
<td>28.555</td>
</tr>
<tr>
<td>fixed ±8 pixels</td>
<td>33.956</td>
<td>32.003</td>
<td>31.959</td>
<td>28.840</td>
</tr>
<tr>
<td>self-adjusting</td>
<td>33.967</td>
<td>31.986</td>
<td>31.970</td>
<td>28.787</td>
</tr>
</tbody>
</table>

Table 3.1 PSNRs of the various test sequences for different bit rate conversions
Table 3.1 demonstrates the improved performance due to the self-adjusting window approach, in terms of PSNR, over the fixed ±1 pixel refinement method. This is also deduced from the results of the fixed ±8 pixels apart from the “Suzie” II and “Foreman” II cases where the successive frames comprise very high degrees of transcoding asymmetry in terms of bit rate conversions. This is indeed an expected result as it is now known that the quality improves with the increasing window size, as stated previously.

Table 3.1 Transcoded bit rates, “Suzie” from 64 kbit/s and “Foreman” from 116 kbit/s on average

<table>
<thead>
<tr>
<th>Bit Rate [kbit/s]</th>
<th>“Suzie” I</th>
<th>“Suzie” II</th>
<th>“Foreman” I</th>
<th>“Foreman” II</th>
</tr>
</thead>
<tbody>
<tr>
<td>fixed ±1 pixel</td>
<td>32.21</td>
<td>23.20</td>
<td>66.04</td>
<td>37.58</td>
</tr>
<tr>
<td>fixed ±8 pixels</td>
<td>33.18</td>
<td>25.20</td>
<td>67.86</td>
<td>42.12</td>
</tr>
<tr>
<td>self-adjusting</td>
<td>32.90</td>
<td>24.56</td>
<td>67.00</td>
<td>40.73</td>
</tr>
</tbody>
</table>

However, as clearly observed from Table 3.2, the bit rate increase due to the refinement has to be kept minimal. This is also achieved with the approach discussed here compared to the fixed ±8-pixel method. Naturally, bit rates obtained from the self-adjustment scheme are slightly higher than those of the fixed ±1 pixel cases as the average variable window sizes are larger than ±1, but smaller than ±8 pixels.

Table 3.2 Transcoded bit rates, “Suzie” from 64 kbit/s and “Foreman” from 116 kbit/s on average

<table>
<thead>
<tr>
<th>No of MBs refined [%]</th>
<th>“Suzie” I</th>
<th>“Suzie” II</th>
<th>“Foreman” I</th>
<th>“Foreman” II</th>
</tr>
</thead>
<tbody>
<tr>
<td>fixed ±1 pixel</td>
<td>23.628</td>
<td>31.572</td>
<td>25.075</td>
<td>42.393</td>
</tr>
<tr>
<td>fixed ±8 pixels</td>
<td>24.283</td>
<td>31.831</td>
<td>27.836</td>
<td>45.997</td>
</tr>
<tr>
<td>self-adjusting</td>
<td>24.406</td>
<td>31.975</td>
<td>27.628</td>
<td>45.363</td>
</tr>
</tbody>
</table>

Table 3.3 Percentage of number of refined MBs to number of overall MBs

Finally, Table 3.3 presents that the number of refined MBs for the self-adjusting window algorithm is more than the fixed ±1 pixel method for highly active scenes, such as “Foreman”, and it is almost similar to the ±1 pixel method for lower activity scenes, such as “Suzie”, resulting in the improved PSNR values for both cases. Consequently, it can be deduced that for low activity scenes the adjustable window reduces in size whereas it expands for high motion activities. This proves that the proposed algorithm successfully tracks the motion activity within a video sequence.

A third dimension of the problem is the added complexity to the scheme. The proposed algorithm clearly offers less complexity than the fixed ±8-pixel method as the number of operations for motion refinement is much less for a smaller window size than for a window of ±8 pixels. Naturally, the complexity is more than the simple ±1 pixel search method.

The experiments were also repeated for other video sequences with different bit rates and various fixed-size search windows and very similar conclusions were derived.
3.5 Frame Rate Reduction

Frame rate reduction is the most widely used method which efficiently allocates additional bits to the remaining frames of the video sequence so as to achieve a satisfactory QoS in a bandwidth-restricted channel. Today's rate controlling algorithms employed in various video coders also adopt this technique in order to comply with the bit rate limitations of the host networks. In cases where the target network cannot support high rates, a video transcoder can also be utilised for frame rate reduction at a video gateway. Moreover, in certain situations when the reduction of the incoming bit rate is not sufficient, bit rate reduction can also be coupled with a frame rate reduction technique. Frame rate reduction can simply be achieved by frame droppings. For instance, if every other frame of a sequence is dropped, this results in a half rate reduction of the original frame rate, e.g. from 30 fr/s down to 15 fr/s.

![Figure 3.16 MV handling during frame rate conversion](image)

In spite of its simplicity, frame rate reduction at the video transcoder causes a mismatch between the video sequence which was originally encoded by the source coder and the same video sequence which was passed through the transcoder. This mismatch can be perceived as annoying for the end-users. If it is not corrected, this distorting effect accumulates throughout the whole video transmission. The degradation in quality is mainly incurred due to the dropped frames, which were originally encoded by the source and hence, are required at the decoder for a successful reconstruction. The importance of the dropped frames for a problem-free reconstruction is further accentuated by the use of predictive decoding. Therefore, whilst reducing the incoming frame rate, the video transcoder also needs to keep track of the motion data. Due to the fact that this information is already received in the input video stream, it is easy for the transcoder to redesign the outgoing MVs from the past input MVs. [Youn3, Youn4] present a simple approach
Chapter 3 Homogeneous Video Transcoding

where the current frame is transcoded by selecting the video frame prior to the first dropped frame as the reference. Even though certain frames are intentionally dropped, the video transcoder still makes use of the motion information from these frames to achieve the required motion data for the frames that are not dropped. This is accomplished by straightforward MV addition. Consequently, the correct motion information can be estimated from the MV logbook situated at the video transcoder.

The outgoing MV set \((T_x, T_y)_n\) is derived from the \((n-k-l)^{th}\) frame as the previous reconstructed reference frame, where \(n\) represents the currently transcoded frame and \(k\) stands for the number of intentionally dropped frames. The sequence of incoming MVs during frame-dropping, \(\{(l_o, l_p)_{n-k}, \ldots, (l_o, l_p)_{n-1}\}\), and the MVs \((l_o, l_p)_n\) from the current frame can be extracted, as shown in Figure 3.16. Thus, the outgoing MVs for the \(n^{th}\) frame are calculated from the extracted MVs as follows:

\[
(T_x, T_y)_n = \left[ \sum_{d=1}^{k+1} (l_x)_{n-d+1}, \sum_{d=1}^{k+1} (l_y)_{n-d+1} \right] 
\]

(3.17)

Furthermore, the newly updated MVs, \((T_x, T_y)_n\), also require refinement, which can be accomplished by a similar MV refinement technique, as described in a previous sub-section.

[Hwang] proposes another method which involves bilinear interpolation. This scheme estimates the MVs of the currently transcoded frame by tracing the motion from the received current frame back to the previous non-skipped frame. However, this is only possible if the MVs between every adjacent frame are known. In this approach, the newly located position, which is based on the interpolated MV, is selected as the new search centre. Thus the MV search range is significantly reduced. The size of the search range is determined by the number of skipped frames and the accumulated magnitudes of their MVs.

### 3.6 Resolution Reduction

Due to its resolution reduction feature, a video transcoder can also be employed for spatial scaleability. This can be accomplished by including a down-sampling filter between the decoder and the re-encoder of the video transcoder [Bjork]. In cases where the end-user terminal is not capable of accommodating the original video frame size, the objective of this filter consists of down-sampling the incoming frame size. This situation occurs when, for example a CIF or larger format video stream is sent to a mobile terminal. Since a mobile terminal is a power-limited device due to the battery life restrictions, it is conventionally built with a small screen. The reduction of the incoming video frame size performed by the transcoder results in the successful
reception of the transmitted stream with the least possible QoS loss. Therefore, this technique provides a very attractive solution for such situations. The loss in the service quality is primarily incurred by the mismatch resulting from the reduction in the resolution of the pictures within the transcoder [Mokry, Vetro]. This distorting effect accumulates throughout the transcoded video, resembling the drift error effect. However, the latter artefact occurs as a result of rate reduction, as explained and demonstrated in Section 3.4.

The example of CIF to QCIF conversion, stated above, requires a resolution reduction by a factor of two in both height and width of the video frame. This reduction leads to a frame size which is a quarter of the original size. As a result, the number of MBs within an original video frame is also quartered. This process introduces the difficulty of choosing the correct MB types and MVs that are required for the candidate outgoing MBs.

![Diagram](image)

**Figure 3.17** Down-sampling by a factor two in each dimension, a- candidate MV selection problem; b- candidate MB type (mode) selection problem

Figure 3.17a demonstrates the first problem which occurs when four MBs are combined to form only one MB. Thus, this operation requires the selection of the most suitable MV out of the four possible MVs. [Bjork] describes a number of simple solutions to this particular problem:

- Averaging the four MVs and dividing the result by two in each dimension.
- Taking the median of any three MVs and dividing the result by two in each dimension.
- Randomly picking one MV out of four and dividing it by two in each dimension.

[Hashe, Hwang, Senda, Shen] propose advanced techniques which yield a more accurate MV selection. However, since the chosen MVs will not be optimal, the MV refinement process is still required.
A similar problem is depicted in Figure 3.17.b, where only one of the four different MB types is passed from the original input video stream to the transcoded output stream. In this case, the problem is more concentrated on choosing the best representative MB type out of the incoming four. A simple solution is described in [Bjork] which proposes a two-step action:

- **Step-1.** If at least one of the four MBs is an intra (I) type, select the I-type. If there is no I-MB but one of the MBs is an inter (P) type, select the P-type. If all four MBs were originally skipped, select the skipped type.

- **Step-2.** The MB types are re-evaluated in the re-encoder following the selection from the first step.

Down-conversion of a CIF video stream to a QCIF stream can also be performed in the coded (DCT) domain, resulting in a significant reduction in the complexity [Zhu]. This approach carries out the DCT motion compensation and applies the down-conversion on an MB-by-MB basis. Thus, four luminance (Y) blocks are reduced to one Y block whilst the chrominance blocks are left unchanged. Once the conversion of four arbitrary MBs is completed, the remaining four chrominance blocks are thereafter reduced to one chrominance block: one individual block for Cb and one for Cr. This method for resolution reduction incurs low complexity due to the fact that it does not require any motion estimation and DCT/IDCT operations. The most important process of the down-conversion algorithm is the DCT motion compensation. It aims to select the most suitable MV for the candidate outgoing MB out of the four incoming original MBs. In addition, [Assun7] introduces a fast computation method for the DCT domain motion compensation algorithm. Other video frame format conversions also comprise similar requirements and follow identical procedures. Furthermore, a fast algorithm for down-scaling an image by a factor of two in the DCT domain was also presented in [Natar]. This resolution reduction technique results in a fairly good transcoded quality with low complexity.

Figure 3.18 A typical resolution transcoding scenario
Chapter 3
Homogeneous Video Transcoding

One possible scenario where this type of a resolution transcoding can be deployed is in a point-to-multipoint video conferencing application, as shown in Figure 3.18. Such a system requires a video gateway which comprises a video transcoder. The objective of the transcoder is to reduce the size of the incoming video stream and distribute the resulting reduced format video streams to several conference attendees. For transmission in the opposite direction, the video gateway multiplexes the smaller format streams into one larger format before transmission.

3.7 Multi-Transcoder Design for Multimedia Traffic Planning

One of the research areas which is gaining considerable amount of interest is multimedia traffic planning. The driving force behind the growing popularity of this area is the integration of several multimedia networks of different characteristics. These differences between the networks inherently introduce limitations on an entire system. One of the most important differences consists of the available bandwidth supported by each network. This limitation imposes a restriction on the number of users supported by the system, resulting in possible congestion scenarios. The latter situation occurs when particularly several users try to gain access to a limited bandwidth [Halsa2]. As a result of congestion, the running applications may be forced to reduce their transmission rates. However, reduced transmission rates inevitably result in reduced QoS levels. This affects each of the network subscribers [Sisod] unless a user prioritised scheme is employed.

Congestion greatly affects all multimedia transmissions, namely speech, audio, data and video information. However, the effects of congestion are mostly conspicuous in video transmissions [Elgeb]. This is due to the fact that during congestion, the streaming VoD data freezes and it can only resume when the system resolves the problem. However, the conventional decoder at the user end has no choice but to skip all the video frames that were frozen during the congestion period. This results in a leap in the video sequence, the duration of which is directly related to the congestion period. The skipped frames introduce QoS degradation. In addition, due to the predictive coding nature of the VoD data, the subsequent video frames transmitted once the congestion problem is resolved, will be wrongly decoded. This is due to the incorrect referencing of the buffer at the decoder end. However, the deterioration in the quality only lasts until the next intra (I) frame is received by the decoder as I-frames refresh the memory of the decoder.

On the other hand, the impacts of congestion on two-way low bit rate applications are much more notable. This is due to the fact that unlike in VoD applications where high bit rate is not
problem, low bit rate services cannot accommodate the additional bits resulting from frequent transmission of I-frames. Therefore, the loss of frames resulting from the congestion period incurs an accumulative prediction error throughout the entire video transmission.

![MPEG-4 Multiparty Video-telephony Scenario](image)

**Figure 3.19** MPEG-4 multiparty video-telephony scenario, using a video transcoder at the MCU for multimedia traffic planning

Figure 3.19 displays a multiparty video telephony/conferencing scenario, which is a very typical two-way low video bit rate communication system. The transcoder simultaneously produces varying outputs at different bit rates for different networks according to their bandwidth requirements, as seen in the figure. The figure depicts a possible real-life scenario where a high quality, high bit rate (R₀) video data is converted to lower bit rates (R₁, R₂) via the video transcoder due to bandwidth and/or congestion problems in these networks. Here, the major achievement of the transcoder is to provide bit rate regulation whilst maintaining the QoS as much as possible. In the case that the two networks at either ends of the transcoder can support same or higher bit rates (R₃, R₄), then the video transcoder maintains this original rate and hence, the original video quality. The video transcoder achieves its objective with the least possible delay and complexity for optimum simplicity. This kind of a transcoder architecture carries out bidirectional operation since it is a two-way gateway structure to video-telephony/conferencing applications.

Since the distinct applications support different transmission rates, it is reasonable to assume an asymmetrical network topology. In this case, a multiparty video communication scenario can only be achieved with the deployment of a video transcoder at the MCU, as depicted in the figure. In the event that the incoming video bitstream has a different rate compared to the rates of the other conference participants, the video transcoder performs rate matching, as explained in the
preceding sections. The aim of this process is to simply alter the original stream in such a way that it can be accommodated by recipient networks. The transformation of bit rates from one network to another, as depicted by this particular scenario, demonstrates a typical example which requires a multimedia traffic planning which in turn can be enhanced by exploiting the useful features of video transcoders [Assunl, Dogan3, Yeado]. The following sub-section presents an application of such a video transcoding algorithm for multimedia traffic planning purposes.

### 3.7.1 The System Architecture for Traffic Planning Purposes

In this sub-section, the video transcoder has been exploited to regulate transmission bit rates for multimedia traffic planning. For this purpose, a bank of video transcoders within an MCU has been designed for the proposed aim [Dogan3]. The bank of video transcoders, as illustrated in Figure 3.20, comprises a series of transcoders combined with each other in parallel, each of which carries out similar operations for the provision of different layers of a video stream. Different layers have different bit rates and quality levels as to suit all various conditions of destination networks. With the use of techniques described in [lanna], the replication of the similar video data in each of the output streams can be optimised for a multi-rate layered multicast video transmission. If a network has a congested node or if any user within a network cannot handle the incoming rate due to bandwidth constraints, such as a mobile-wireless network, the transcoder bank reduces the original rate. For networks that can support higher rates or at least the incoming rate, the transcoder bank can also maintain the incoming rate with the original quality without the need for transcoding the input video data, as shown in Figure 3.20.

![Figure 3.20 System architecture; the video transcoder bank (superscripts i and o representing the inputs and outputs, respectively and subscripts representing the network types)](image)
It has also been considered to employ only one video transcoder producing simultaneous multi-rate outputs, for a typical point-to-multipoint video-conferencing scenario. However, this has not been implemented in this research due to its high complexity requirement and long processing delays, which cannot be tolerated by two-way video transmissions. As an alternative, a design of more than one video transcoder operating in parallel, as in Figure 3.20, is thought to be a more efficient and less delay-prone implementation as the processing delay will only be one transcoding cycle.

With the use of a video transcoder bank, separate layers of individual compressed video data copies can be produced for a particular point-to-multipoint operation. In the case that more inputs are considered, as in a multipoint-to-multipoint video transmission scenario, then the transcoder bank is made capable of handling more than one input at a time. This is more like a time-shared networking issue between different inputs and hence, it should be considered by the MCU itself.

Even though it seems like certain transcoders are dedicated to fixed output bit rates from Figure 3.20, a further flexible and adaptive operation has also been introduced to the system for time varying bandwidth requirements. For instance, if network-2 is congested at a time, as stated earlier, and if the congestion is resolved before the video transmission ceases, then there is no need to continue to supply the lower rate to that particular network. In such a case, network congestion feedback can tell the associated video transcoder to increase the rate and hence, also increasing the QoS. Conversely, if the congestion conditions persist, the feedback report is signalled to the transcoder to further reduce the output rate. Thus, each individual video transcoder within the bank operates in an adaptive way to the new conditions and relays the most suited output with the new rate. This feedback signalling, which reports the varying network congestion conditions, can simply be incorporated using a buffering mechanism. The buffers at the end of each transcoder estimate the congestion characteristics of the host networks by using the occupancy factor. Thus, according to the buffer occupancy, the feedback signal reports back to the particular video transcoder to increase or decrease the output rate.

Moreover, as the original rate output (upper path in the figure) can be sent without any transcoding operation and hence, its associated processing delay, the synchronisation of this particular stream has to be maintained with the other layers by the MCU. Thus, the MCU is required to accomplish a synchronisation process between different outputs of the transcoder bank.
Chapter 3 Homogeneous Video Transcoding

3.8 Computer Simulations and Analysis of the Results

In this section, the computer simulation results are presented to verify the theoretically discussed transcoding schemes. Similarly, [Gopal, Tudor, Yeado] have investigated the performance of real-time operation of homogeneous video transcoders whilst also giving the experimental results. The results shown here are provided from an exhaustive number of simulations for different video sequences and bit/frame rates. Both the objective and subjective simulation results are demonstrated in this section.

3.8.1 Bit Rate Reduction

3.8.1.1 Experiments and Results

The bit rate reduction tests were performed for an exhaustive number of experiments for numerous video test sequences and various rate conversion combinations. Here, only the results of a selection of these experiments have been presented. Thus, the results are given and discussed for the standard video test sequences, such as “Suzie”, “Carphone”, “Foreman”, “Akiyo_with_Crowd” and “Sign_Irene” in an increasing motion activity order. The number of transcoded frames and the rate down-conversion levels are given within the experimental sets which were carried out with a frame rate of 25 fr/s and in I-P-P-P-P-... format for QCIF (176x144 pixels) resolution.

The first set of results introduces the average bit rates and PSNRs of the open-loop transcoding method in Figure 3.21. The numbering in the legends refers to the upper portion of Table 3.4. The results comprise open-loop transcoding of the 150-frame “Suzie” sequence. In these results, the effects of the rate reduction on the first I-frame are depicted. Moreover, they also show the results when the motion data is ignored and taken into consideration during the selection of the skipped mode for the MBs whilst transcoding. For this purpose, a control mechanism is employed such that an MB which has all zero DCT coefficients, due to the coarser quantisation process during transcoding, is not directly set as a skipped MB unless it has any motion data associated with it. This particular point has not been addressed in any previous work.

The second set of results, shown in Figure 3.22 and the lower portion of Table 3.4, presents the outcome of the closed-loop transcoding experiments where drift effect of the open-loop transcoding was compensated. Among these results, the effect of the first I-frame transcoding has also been presented, in terms of bit rate and PSNR reductions.

Moreover, the first two sets, namely open- and closed-loop transcoding results, are compared to the reference results which were originally encoded and decoded at the required bit rates.
Figure 3.21 Objective test results for the open-loop video transcoding of the “Suzie” sequence, \( a \)- PSNR and \( b \)- bit rate variations

![Figure 3.21](image)

Figure 3.22 Objective test results for the closed-loop video transcoding of the “Suzie” sequence, \( a \)- PSNR and \( b \)- bit rate variations

![Figure 3.22](image)

<table>
<thead>
<tr>
<th>Type of the Scheme</th>
<th>Av. Bit Rate [kb/s]</th>
<th>Av. PSNR [dB]</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) direct enc/dec @ QP=8</td>
<td>62.015</td>
<td>36.803</td>
</tr>
<tr>
<td>(2) direct enc/dec @ QP=14</td>
<td>35.323</td>
<td>34.619</td>
</tr>
<tr>
<td><strong>Open-loop video transcoding</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(3) with motion care &amp; w/out I-frm transcoding</td>
<td>27.588</td>
<td>27.702</td>
</tr>
<tr>
<td>(4) with motion care &amp; I-frm transcoding</td>
<td>26.495</td>
<td>27.509</td>
</tr>
<tr>
<td>(5) w/out motion care &amp; w/out I-frm transcoding</td>
<td>22.613</td>
<td>22.300</td>
</tr>
<tr>
<td><strong>Closed-loop video transcoding</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(6) w/out I-frm transcoding</td>
<td>32.723</td>
<td>33.850</td>
</tr>
<tr>
<td>(7) with I-frm transcoding</td>
<td>31.775</td>
<td>33.603</td>
</tr>
</tbody>
</table>

Table 3.4 The effects of the I-frame transcoding and the motion data care during open-loop transcoding
Finally, Figures 3.23 and 3.24 demonstrate various objective and subjective rate conversion results for different test sequences, respectively. In addition, Table 3.5 illustrate the detailed bit rates and PSNRs associated with the results given in the preceding figures.
Figure 3.23 Objective test results for the homogeneous video transcoding of the various video sequences, a- “Suzie” (QP=8→14) (I- PSNR, 2- bit rate); b- “Foreman” (QP=8→14) (I- PSNR, 2- bit rate); c- “Sign_Irene” (QP=7→17) (I- PSNR, 2- bit rate); d- “Carphone” (QP=8→14); e- “Akiyo_with_Crowd” (QP=5→12); f- “Foreman” (QP=20→30); g- “Suzie” (QP=16→23)
Figure 3.24 Subjective results of the last frames of the “Suzie”, “Foreman”, “Carphone”, “Sign_Irene” and “Akiyo_with_Crowd” sequences, a- direct enc/dec (at higher bit rates specified in Table 3.5); b- direct enc/dec (at lower bit rates specified in Table 3.5); c- re-quantisation scheme (1- motion data is ignored, 2- motion data is not ignored); d- cascaded fully dec/re-enc scheme; e- MV re-use scheme; f- MV re-estimation scheme; g- MV refinement scheme

<table>
<thead>
<tr>
<th>Type of the Scheme</th>
<th>Av. Bit Rate [kbit/s]</th>
<th>Av. PSNR [dB]</th>
</tr>
</thead>
<tbody>
<tr>
<td>150-frame “Suzie” @ 25 fps (QP=8→14); rate reduction=50%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(1) direct enc/dec @ QP=8</td>
<td>62.915</td>
<td>36.803</td>
</tr>
<tr>
<td>(2) direct enc/dec @ QP=14</td>
<td>35.323</td>
<td>34.619</td>
</tr>
<tr>
<td>(3) cascaded fully dec/re-enc</td>
<td>34.020</td>
<td>34.038</td>
</tr>
<tr>
<td>(4) transc MV re-ux (closed-loop)</td>
<td>32.723</td>
<td>33.850</td>
</tr>
<tr>
<td>(5) transc re-quant (open-loop)</td>
<td>27.588</td>
<td>27.702</td>
</tr>
<tr>
<td>(6) transc MV re-estimate: ±16 pels</td>
<td>33.760</td>
<td>34.019</td>
</tr>
</tbody>
</table>
Chapter 3  Homogeneous Video Transcoding

Table 3.5 Bit rate transcoding results for various video test sequences

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Rate Reduction (%)</th>
<th>Direct Enc/Dec</th>
<th>Direct Enc/Dec @ QP=14</th>
<th>Cascaded Fully Dec/Enc</th>
<th>Transcoded MV Re-use (Closed-Loop)</th>
<th>Transcoded MV Re-Quant (Open-Loop)</th>
<th>Transcoded MV Re-estimate ±16 Pels</th>
<th>Transcoded MV Refinement ±3 Pels</th>
</tr>
</thead>
<tbody>
<tr>
<td>200-frame “Foreman” @ 25 fr/s (QP=8 → 14); rate reduction=50%</td>
<td></td>
<td>116.663</td>
<td>61.970</td>
<td>58.035</td>
<td>57.443</td>
<td>43.473</td>
<td>58.180</td>
<td>57.830</td>
</tr>
<tr>
<td>300-frame “Carphone” @ 25 fr/s (QP=8 → 14); rate reduction=55%</td>
<td></td>
<td>117.795</td>
<td>58.760</td>
<td>53.433</td>
<td>59.483</td>
<td>44.145</td>
<td>53.440</td>
<td>52.975</td>
</tr>
<tr>
<td>500-frame “Sign_Irene” @ 25 fr/s (QP=7 → 17); rate reduction=70%</td>
<td></td>
<td>111.625</td>
<td>32.238</td>
<td>33.663</td>
<td>32.720</td>
<td>26.128</td>
<td>33.715</td>
<td>33.353</td>
</tr>
<tr>
<td>300-frame “Akiyo with Crowd” @ 25 fr/s (QP=5 → 12); rate reduction=66%</td>
<td></td>
<td>357.400</td>
<td>128.753</td>
<td>125.483</td>
<td>125.395</td>
<td>87.075</td>
<td>125.730</td>
<td>125.580</td>
</tr>
<tr>
<td>200-frame “Foreman” @ 25 fr/s (QP=20 → 30); rate reduction=17%</td>
<td></td>
<td>47.193</td>
<td>39.775</td>
<td>39.470</td>
<td>39.745</td>
<td>37.765</td>
<td>39.325</td>
<td>40.410</td>
</tr>
<tr>
<td>150-frame “Suzie” @ 25 fr/s (QP=16 → 23); rate reduction=16%</td>
<td></td>
<td>31.823</td>
<td>27.580</td>
<td>27.230</td>
<td>26.458</td>
<td>24.489</td>
<td>27.103</td>
<td>26.468</td>
</tr>
</tbody>
</table>

Table 3.5 Bit rate transcoding results for various video test sequences
3.8.1.2 Analysis of the Results

I-frame transcoding results have presented an average of 0.2 dB quality loss compared to the case where this particular frame was preserved during the transcoding process. However, it also gave an average of 1 kbit/s less bit rate for both the open- and closed-loop transcoding operations when Figures 3.21-22 and Table 3.4 are examined. On the other hand, as the results clearly depict, the motion ignorance during the open-loop transcoding gave an approximate of 5 dB loss in transcoding quality. Thus, for the rest of the experiments, the video transcoder was designed to consider the existence of the possible motion data even though an MB contained all zero coefficients. This particular control for the motion information gave 5 dB better quality at the expense of an average of 5 kbit/s increase in the bit rate. These results were obtained for a particular “Suzie” down-conversion process whereas similar results were also derived, but not presented here, for different test sequences. The subjective effects of this outcome, which are illustrated in Figures 3.24.c1 and c2, prove that this kind of a motion check was worth employing.

Following this initial experimentation, Figure 3.23 and Table 3.5 demonstrate the average bit rate and PSNR results for various rate down-scaling processes. From these results, it can be observed that between the open-loop and closed-loop video transcoding, there is a quality gap of 4 to 12 dB in terms of PSNR in favour for the closed-loop operations. However, the gain obtained with the employment of the drift correction algorithms is counteracted by a bit rate increase of 2 to 38 kbit/s on average. It is worth mentioning that the quality gain and the associated bit rate increase are directly proportional to the amount of motion activity of the scene and the rate down-conversion proportion. Thus, the higher the rate down-scaling proportion and/or the higher the motion activity within a particular video clip, the better the quality compensation as opposed to the drift-prone applications.

Furthermore, the results also showed that the four drift-free tandem methods, namely conventional cascaded decoding/re-encoding, motion data re-use, motion data re-estimate and motion data refinement schemes, performed similar to each other in terms of the bit rate and PSNR values. However, as explained in the previous sections, the algorithmic complexities do not show similar behaviours for different transcoding schemes. This makes the MV refinement scheme become the most adequate method amongst the others regarding the provision of reasonably good quality and moderate complexity associated with it. The resulting subjective performances can be seen in Figure 3.24 for the different video sequences and rate conversion asymmetries.

Finally, it can also be noted from the results presented in Table 3.5 that highly active video sequences, in terms of motion data, and high asymmetry of rate conversions require increasing MV search window sizes for the refinement purposes. This has also been discussed in detail in the preceding sections and verified one more time here.
3.8.2 Frame Rate Reduction

3.8.2.1 Experiments and Results

In the experiments to investigate the reduction performance of the original input frame rate of a video transmission, five different video sequences were tested. These video sequences were chosen accordingly representing different motion activity natures. From moderate to high activity QCIF size (176x144 pixels) “Claire”, “Suzie”, “Carphone”, “Foreman” and “Akiyo_with_Crowd” were originally encoded at average bit rates of 68.7, 94.5, 56.5, 87.4 and 161.5 kbit/s with frame rates of 25 fr/s in I-P-P-P-P-... format, respectively. Each of the test sequences were then transcoded down to half of the incoming original frame rate, which resulted in an eventual overall transmission rate reduction. Therefore, the resulting qualities due to the frame rate re-scaling were also compared to the qualities of the same video clips with a sole bit rate reduction process preserving the original frame rates. The sole bit rate reductions were accomplished using the drift-free MV refinement transcoding scheme. These final experiments were carried out to verify the performance evaluation of the frame rate reduction algorithm. Moreover, frame rate transcoding operations were performed with and without the use of an MV logbook within the transcoder. The MV logbook contains the stored MVs of the intentionally dropped video frames. The experimental results are presented in Table 3.6. The total numbers of transcoded video frames and the MV refinement window sizes are also shown within the table for each of the different test sequences.

<table>
<thead>
<tr>
<th>Type of the Scheme</th>
<th>Av. Bit Rate [kbit/s]</th>
<th>Frame Rate [fr/s]</th>
<th>Quantisation Parameter [QP]</th>
<th>Av. PSNR [dB]</th>
</tr>
</thead>
<tbody>
<tr>
<td>direct enc/dec</td>
<td>68.760</td>
<td>25</td>
<td>4</td>
<td>41.56</td>
</tr>
<tr>
<td>fr-rate trnsed only</td>
<td>14.686</td>
<td>25 → 12.5</td>
<td>4 → 10</td>
<td>36.688</td>
</tr>
<tr>
<td>(without MV add.)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>fr-rate trnsed only</td>
<td>14.741</td>
<td>25 → 12.5</td>
<td>4 → 10</td>
<td>36.682</td>
</tr>
<tr>
<td>(with MV add.)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>b-rate trnsed only</td>
<td>14.370</td>
<td>25 → 25</td>
<td>4 → 15</td>
<td>35.509</td>
</tr>
</tbody>
</table>

200-frame “Claire”, MV refinement window size: ±2 pixels

<table>
<thead>
<tr>
<th>Type of the Scheme</th>
<th>Av. Bit Rate [kbit/s]</th>
<th>Frame Rate [fr/s]</th>
<th>Quantisation Parameter [QP]</th>
<th>Av. PSNR [dB]</th>
</tr>
</thead>
<tbody>
<tr>
<td>direct enc/dec</td>
<td>94.575</td>
<td>25</td>
<td>6</td>
<td>38.171</td>
</tr>
<tr>
<td>fr-rate trnsed only</td>
<td>58.233</td>
<td>25 → 12.5</td>
<td>6 → 6</td>
<td>35.852</td>
</tr>
<tr>
<td>(without MV add.)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>fr-rate trnsed only</td>
<td>56.212</td>
<td>25 → 12.5</td>
<td>6 → 6</td>
<td>35.920</td>
</tr>
<tr>
<td>(with MV add.)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>b-rate trnsed only</td>
<td>56.623</td>
<td>25 → 25</td>
<td>6 → 8</td>
<td>35.862</td>
</tr>
</tbody>
</table>

150-frame “Suzie”, MV refinement window size: ±2 pixels

<table>
<thead>
<tr>
<th>Type of the Scheme</th>
<th>Av. Bit Rate [kbit/s]</th>
<th>Frame Rate [fr/s]</th>
<th>Quantisation Parameter [QP]</th>
<th>Av. PSNR [dB]</th>
</tr>
</thead>
<tbody>
<tr>
<td>direct enc/dec</td>
<td>56.518</td>
<td>25</td>
<td>12</td>
<td>33.609</td>
</tr>
<tr>
<td>fr-rate trnsed only</td>
<td>32.534</td>
<td>25 → 12.5</td>
<td>12 → 12</td>
<td>30.500</td>
</tr>
<tr>
<td>(without MV add.)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>fr-rate trnsed only</td>
<td>32.840</td>
<td>25 → 12.5</td>
<td>12 → 12</td>
<td>30.610</td>
</tr>
<tr>
<td>(with MV add.)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>b-rate trnsed only</td>
<td>32.402</td>
<td>25 → 25</td>
<td>12 → 20</td>
<td>30.608</td>
</tr>
</tbody>
</table>

200-frame “Carphone”, MV refinement window size: ±3 pixels

<table>
<thead>
<tr>
<th>Type of the Scheme</th>
<th>Av. Bit Rate [kbit/s]</th>
<th>Frame Rate [fr/s]</th>
<th>Quantisation Parameter [QP]</th>
<th>Av. PSNR [dB]</th>
</tr>
</thead>
<tbody>
<tr>
<td>direct enc/dec</td>
<td>87.403</td>
<td>25</td>
<td>10</td>
<td>33.582</td>
</tr>
</tbody>
</table>
Chapter 3 Homogeneous Video Transcoding

Table 3.6 Frame rate transcoding results for various video test sequences

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Frame Rate</th>
<th>B-Rate</th>
<th>Spatial Quality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct Encode</td>
<td>161.580</td>
<td>25</td>
<td>32.690</td>
</tr>
<tr>
<td>Fr-rate trncd only</td>
<td>76.148</td>
<td>25</td>
<td>29.802</td>
</tr>
<tr>
<td>Fr-rate trncd only (with MV add.)</td>
<td>76.351</td>
<td>25</td>
<td>29.815</td>
</tr>
<tr>
<td>B-rate trncd only</td>
<td>77.123</td>
<td>25</td>
<td>28.737</td>
</tr>
</tbody>
</table>

3.8.2.2 Analysis of the Results

Table 3.6 presents two different sets of experiments: one of which conducted only frame rate reduction and the other performed bit and frame rate reductions at the same instant. This can be observed from the changing quantisation parameter within an experiment. However, regardless of the test set-ups, mere bit rate reduction operations required much coarser quantisation step-sizes than their counterpart frame rate transcoding operations as to achieve similar output transmission rates from the same inputs. Yet from the table, it can be noted that the frame rate reduction gave a slightly better spatial quality than the bit rate reduction at the same bit rate. The quality improvement is averaged to be 0.5 dB, with a maximum value of 1 dB for both “Claire” and “Akiyo_with_Crowd” sequences. This is due to the fact that the frame rate reduction allows the transcoder to efficiently allocate the bits for the remaining frames. Thus, frame rate reduction allocates more bits per frames as the number of frames is reduced. Therefore, a certain quality level is preserved. However, various simulation results have shown that the improved PSNRs are only observed when there is relatively high motion activity within the video scenes. In other situations, both cases provide similar levels of quality. This is an important outcome as the combination of the bit and frame rate transcoding schemes supports lower transmission rates with acceptable qualities. On the other hand, the temporal qualities of the transcoded video streams suffer from frame dropping effects with the increasing sequence numbers as the MV estimation is carried out with reference only to the remaining video frames in the sequences. This imposes sudden jumps with degraded motion quality in the video streams to the viewer resulting in less number of output video frames and hence, decreased frame rate. Since every other video frame is regularly skipped to halve the input frame rate, jitter effect is not seen in the transcoded streams whereby channel conditions do not vary with time either, such as observed in the course of the simulations.

In addition, Table 3.6 also presents the results for frame rate transcoding without the MV addition scheme. Transcoding without the use of MV data, which belongs to the intentionally dropped
frames, achieved slightly worse quality than the one using an MV logbook. This is due to the fact that the transcoder makes use of the available motion data of the dropped video frames to increase the output quality, as explained earlier. On the other hand, this correction factor incurs a small degree of bit rate increase which is imposed by the fact that input original MVs are elaborated for a better suited output motion information. Referring to the table, the average increase in the resulting bit rate was experienced to be 0.2 kbit/s, with a maximum of 1.6 kbit/s for the “Foreman” sequence.

### 3.8.3 Multimedia Traffic Planning

#### 3.8.3.1 Experiments and Results

Simulations were performed using the “Foreman” video test sequence with 200 frames and the “Suzie” sequence with 150 frames. “Foreman” and “Suzie” were originally encoded at 116 kbit/s and 64 kbit/s on average with a frame rate of 25 fr/s in I-P-P-P-P-... format, respectively. The sequences were encoded, transcoded and decoded in compliance with the MPEG-4 standard with the use of unrestricted MVs and advanced prediction modes. The frame resolutions were chosen as QCIF with 176×144 pixels. The results given in the following sets of figures and the table comprise the PSNR and the bit rate variations versus the number of transcoded frames.

Figure 3.25 demonstrates the different quality levels of the multi-transcoder architecture, namely the video transcoder bank, at various bit rates for “Foreman”. The multi-rate outputs are generated from a single rate input to the video transcoder bank. Table 3.7 presents the detailed bit rate and PSNR results for individual rates.

![Figure 3.25 PSNRs of the layered transcoder bank outputs for “Foreman”](image-url)
Table 3.7 Average bit rate and PSNR values for Figure 3.25

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$R_0$</td>
<td>116.063</td>
<td>34.642</td>
<td>N/A</td>
</tr>
<tr>
<td>$R_1$</td>
<td>99.208</td>
<td>33.531</td>
<td>±1</td>
</tr>
<tr>
<td>$R_2$</td>
<td>82.088</td>
<td>32.755</td>
<td>±1</td>
</tr>
<tr>
<td>$R_3$</td>
<td>66.038</td>
<td>31.947</td>
<td>±1</td>
</tr>
<tr>
<td>$R_4$</td>
<td>57.830</td>
<td>31.380</td>
<td>±3</td>
</tr>
<tr>
<td>$R_5$</td>
<td>52.778</td>
<td>30.891</td>
<td>±4</td>
</tr>
<tr>
<td>$R_6$</td>
<td>45.218</td>
<td>29.792</td>
<td>±6</td>
</tr>
<tr>
<td>$R_7$</td>
<td>41.655</td>
<td>28.827</td>
<td>±7</td>
</tr>
</tbody>
</table>

Figure 3.26 Objective results of the "Foreman" sequence for the adaptive multimedia traffic planning, a- PSNR and b- bit rate variations

Figure 3.27 Objective results of the "Suzie" sequence for the adaptive multimedia traffic planning, a- PSNR and b- bit rate variations
Finally, Figures 3.26 and 3.27 present the PSNR and bit rate results of the proposed adaptive congestion control technique for the “Foreman” and “Suzie” sequences, respectively. As noticed from the figures, channel congestion characteristics varied throughout the video transmission simulations. The instantaneous bit rate and PSNR changes from the default values are given within the figures, (-) sign representing the decrease and (+) sign showing the increase from the fixed rate levels. The PSNR figures also present the results for certain fixed rate outputs of the transcoder bank, which fit in the varying congestion requirements, for comparison purposes.

3.8.3.2 Analysis of the Results

It can be observed from Figure 3.25 that as the output rate decreases, so does the video quality. However, it is also notable that for an approximately 46 kbit/s decrease from $R_i$ to $R_5$, the transcoded video quality only degrades for about 2.5 dB on average, which is fairly imperceptible to human eye subjectively. Furthermore, Table 3.7 proves one more time that higher rate reduction asymmetry during transcoding requires larger MV refinement window sizes for a better QoS provision.

The particular channel presented in Figure 3.26 already had congestion. Thus, the video transcoder bank produced a reduced rate, $R_2 = 82$ kbit/s on average, from the input rate of $R_0 = 116$ kbit/s on average. However, due to the additional congestion, the supported rate within the network reduced down to 66 and 53 kbit/s on average, respectively. This is evident from the figure for periods of the 50th-100th and the 150th-175th video frames. For these particular cases, video transcoder yielded reduced rates as required by the congestion bottleneck yet at the cost of a quality degradation. On the other hand, between the 110th and 125th frames, the already existing congestion condition was resolved for a short time. As a result, the adaptive transcoder bank increased the bit rate providing the best achievable quality for this short time period. As clear from the figures, the transcoder bank follows the variation of congestion conditions adaptively with the help of feedback signalling. In these simulations, since the congestion monitoring feedback signal is received from the output buffer of the transcoding bank, the arrival of this particular control signal back at the adaptive multi-transcoder system was not assumed to be significantly delayed. However, a succeeding chapter will also introduce similar scenarios where the feedback signal is received from an end-node and hence, further delayed. Therefore, Chapter 5 will also be discussing the resulting effects of the frame losses due to congestion and/or severe error conditions and methods to alleviate these problems where the feedback control signal has a certain amount of delay.

Referring to the obtained results, the proposed adaptive scheme gave a better overall PSNR level (32.483 dB at 78 kbit/s on average) showing that the video quality degraded only for a short time
with this technique until the congestion was resolved. On the contrary, fixed low rates exhibited continuous low quality measures regardless of the varying congestion conditions.

Moreover, the adaptive multimedia traffic planning results for “Suzie” present a fairly similar behaviour as obtained for the “Foreman” sequence, as observed from Figure 3.27. The results depict that the output of the multi-transcoder architecture responded adequately to the varying congestion conditions during the overall transmission for the intervals between 20\textsuperscript{th}-40\textsuperscript{th}, 70\textsuperscript{th}-80\textsuperscript{th} and 115\textsuperscript{th}-130\textsuperscript{th} video frames. The video transcoder bank output, at an average bit rate of 32 kbit/s down from an original rate of 64 kbit/s, was adaptively changed down to 30 and 26 kbit/s and also up to 56 kbit/s depending on the altering simulated channel bottleneck situations. These last simulation results also showed that the adaptive scheme with the use of a congestion-adaptive multi-transcoder architecture gave a better overall PSNR level (33.902 dB at 32 kbit/s on average) compared to the fixed low rate outputs with continuous low qualities.

### 3.9 Concluding Remarks

One of the important requirements in networking scenarios is interoperability. Thus, in this chapter, it has been discussed that video transcoding is an important interoperability operation, which is transparent to the end-user, for a seamless interconnection of diverse multimedia networks and standards.

In heterogeneous networking topologies, which comprise multiple systems operating at different rates, the transmission rate adjustment can sometimes be crucial for successful reception. Therefore, rate management algorithms are required at certain gateways that accomplish necessary rate matching operations between particular networks. The research described in this chapter has been dedicated for the investigation of such operations. It has also been conducted to establish the basics of an integral low bit rate video traffic management scheme. In this direction, rate conversion algorithms have been classified into two major groups: bit rate and frame rate reduction techniques. In addition, bit rate reduction methods are gradually developed to a final stage from the conventional and the most primitive application of all, namely cascaded fully decoding and re-encoding scheme. During this elaboration, one specific transcoding scheme has been built on the basis of its predecessor but bearing superior features, which also imposed increasing computational complexity. However, the ultimate scheme that has been developed and later widely exploited, namely the transcoding scheme using motion data refinement, has the optimised complexity and comparable quality characteristics. Table 3.8 below is a vintage of this development which clearly summarises the advantages and disadvantages of each of the step-stone transcoding schemes.
### Table 3.8 Summary of the advantages/disadvantages of the five bit rate transcoding schemes

<table>
<thead>
<tr>
<th>Type of the Scheme</th>
<th>Extra Operations Required</th>
<th>Complexity/Delay</th>
<th>Output Quality</th>
</tr>
</thead>
<tbody>
<tr>
<td>cascaded fully decoding/re-encoding</td>
<td>frame re-order &amp; full-scale (±16 pels) MV re-estimate</td>
<td>highest complexity longest delay</td>
<td>moderate quality images</td>
</tr>
<tr>
<td>transcoding with re-quantisation</td>
<td>N/A</td>
<td>lowest complexity shortest delay</td>
<td>very low quality images due to the drift effect</td>
</tr>
<tr>
<td>transcoding with MV re-use</td>
<td>drift correction loop addition</td>
<td>2nd lowest complexity 2nd shortest delay</td>
<td>moderate quality images</td>
</tr>
<tr>
<td>transcoding with MV re-estimation</td>
<td>drift corr. loop &amp; full-scale (±16 pels) MV re-estimate</td>
<td>2nd highest complexity 2nd longest delay</td>
<td>moderate quality images</td>
</tr>
<tr>
<td>transcoding with MV refinement</td>
<td>drift corr. loop &amp; small-scale (±1-4 pels) MV refine</td>
<td>moderate complexity moderate delay</td>
<td>slightly higher quality images</td>
</tr>
</tbody>
</table>

The vast number of experiments conducted during the simulations of the different transcoding schemes have exhibited that drift is a significant QoS degrading effect which is supposed to be stopped. It has also been shown that the occurrence of the drift can be overcome by a drift correction loop. Moreover, with the addition of an MV refinement algorithm, more suited output MVs can be designed to give noticeably increased quality levels. The numerous experiments presented that a video transcoder featuring a drift correction loop and an MV refinement block gave an average of 12 dB better quality than the drift-prone algorithms.

Similarly, frame rate reduction algorithms have been designed and tested. Promising results for the support of further reduced transmission rates have been obtained to warrant very low bit rates which cannot merely be achieved by the sole use of bit rate reduction methods. Furthermore, it has also been presented that an additional MV logbook, which keeps track of the motion data of the intentionally dropped video frames, favours the transcoding quality.

In this chapter, it has also been demonstrated that an MPEG-4 video transcoder bank can accomplish the necessary bit rate management for multimedia traffic planning. This transcoding operation has been achieved with the best available QoS and minimal complexity. The results yielded necessary bit rates for the particular demanding networks at reasonable video qualities. Moreover, an adaptive scheme to monitor and control the varying congestion conditions with a feedback loop has also been employed. This kind of an achievement has given a better overall service quality as it exploited the changing conditions and available bandwidth more efficiently than a fixed rate output.

Furthermore, preliminary results showed that the increase in MV refinement search window size resulted in a slightly improved performance yet at the expense of a higher bit rate and complexity. Meanwhile, this rather precious piece of information has been exploited and a self-adjusting motion refinement search window approach has been developed to achieve a fine-tuning of video transcoding quality. The automatic refinement window has been proved to trace the motion activity of a video sequence to enlarge for high activity scenes and shrink for low activity ones.
This scheme has also been presented to give an optimised bit rate and complexity over conventional techniques where fixed-size search windows are employed for motion refinement.

Lastly, as to summarise the novelty introduced in this particular chapter, the following achievement can be mentioned:

* The design and experimentation of an MPEG-4 compatible homogeneous video transcoding algorithm for bit and frame rate conversions.
* A full-scale rate reduction transcoding study comprising different conventional and contemporary schemes.
* The introduction of a control mechanism which checks for the existence of any motion information during the drift correction process.
* The development of a methodology to accomplish a self-adjusting MV refinement window size scheme for a finer transcoding quality tuning achievement.
* The design of a novel efficient multimedia traffic management algorithm comprising a bank of video transcoders operating in parallel to each other.
* Further optimisation of the management algorithm with the addition of feedback controlling within the video transcoder bank to monitor the varying congestion conditions in a network.
Chapter 4

4 Heterogeneous Video Transcoding

4.1 Introduction

The delivery of video data over various network platforms requires a successful interconnection of such networks. Video communications demand a seamless delivery of video content to a broad range of users with different bandwidth and resource constraints. However, the video attributes and compression standards used by source coders, the communication channels and the client devices are generally far from being transparent in nature. Thus, the design of an efficient video transcoding algorithm is required to compensate for these mismatches and provide the users with the seamless experience. Video transcoding is essential for end-to-end compatibility of two or more different networks operating with different characteristics and constraints. The aim is to resolve the interoperability problem and provide a global interconnection of different network topologies.

In the light of these facts, the interoperability issue is discussed in this chapter along with the seamless interconnection of two low bit rate video compression standards, namely H.263 and MPEG-4. Thus, an MPEG-4 to/from H.263 video transcoder enables the transmission of the MPEG encoded video data over channels which support H.263, such as ISDN lines, analogue telephone (PSTN) lines, wireless links and other low bandwidth network channels.

This chapter is organised as follows: The second section presents an overview of the heterogeneous video transcoding. The third section discusses the importance of using the two different video compression standards, namely H.263 and MPEG-4, and also their operation in tandem in asymmetric networks scenarios. The fourth section focuses on different tandem structures and the major types of MPEG-4/H.263 heterogeneous transcoding issues. The fifth section demonstrates the computer simulations and their results. Moreover, it also presents the discussions on the performance of a novel bi-directional video transcoder implementation which fully connects different networks employing the H.263 and MPEG-4 standards operating at low bit rates. Finally, the sixth section draws the concluding remarks.
4.2 Heterogeneous Video Transcoding

The importance of interoperability between diverse network structures has been addressed in the introduction. The second major type of video transcoding algorithms comprises features which provide a tandem operation for different network topologies. Tandem is the linking or interconnection of two or more video compression standards. This second type of transcoding is referred to as heterogeneous video transcoding. This kind of algorithm involves a method of tandem in which a translation of the syntax of two standards takes place without the need for any further decoding and re-encoding processes. Therefore, the heterogeneous video transcoding accomplishes an algorithm conversion whereby a coded bitstream format of one standard is successfully translated to a different standard, without the performance of a conventional cascaded decoding/re-encoding.

The main objective of the heterogeneous video transcoding is to reduce the processing power and the time delay whilst improving the QoS. This is achieved by eliminating the distortion caused by the cascaded decoding and re-encoding processes which exist at the interconnection point of two different video standards. This interconnection point can be a video gateway including a heterogeneous video transcoder architecture, as depicted in Figure 4.1. Since the video transcoder only translates and maps the syntax of the two video formats, it does not accomplish any complex and time consuming motion estimation, motion compensation and DCT/IDCT processes. Thus, the complexity is significantly reduced resulting in a shorter processing delay. This is extremely desirable, particularly in long round-trip delay which is a characteristic of communications over satellite networks and also in delay sensitive applications, such as two-way videotelephony/conferencing.

![Figure 4.1 Simplified block diagram of a tandem of the two video communication standards](image)

This chapter concentrates on the heterogeneous video transcoding issues due to the importance of the interoperability of different video coding standards. With the frequent addition of new and diverse multimedia networks to the communication systems, the interoperability problem has
recently started to draw a significant attention from the research point of view. Moreover, the latest 3G and the forthcoming beyond 3G mobile communications experts have declared that the interoperability of diverse media is the primary issue to be addressed. The increasing number of publicity in the literature also proves that the interoperability has recently become important. Among these papers, [Dogan2] introduces the uni-directional and [Dogan4] presents the bi-directional interoperability techniques between H.263 and MPEG-4 video standards at low bit rates. On the other hand, [Shana1, Shana2, Shana3] discuss the high to low bit rate standard conversion mechanism between MPEG-1/2 and H.261/H.263. In addition, [Iwasa] carries out experiments of MPEG-4 to H.261 video transcoding in a recent paper. [Feams] takes the MPEG-2 to H.263 video transcoder one step further by accomplishing the transcoding of an interlaced MPEG-2 bitstream to a lower bit rate progressive H.263 bitstream. This algorithm presents a field to frame conversion with the associated rate reduction and spatial down-sampling operations. Furthermore, [Wu] introduces a method for transcoding JPEG pictures into MPEG-1 video. Conversely, a fast compressed domain MPEG-1 video to motion-JPEG (M-JPEG) [Lei] transcoding algorithm without fully decompressing the MPEG-1 source has also been demonstrated in [Achar].

4.3 MPEG-4/H.263 Video Transcoding

In addition to communications between two wireless video-phones, there are also demands for communications between wireless and wired systems (i.e. video-conferencing or video-phone systems over ISDNs, PSTNs). Communications between these two different systems require transcoding as each system is characterised by different coding formats. The current wired video communications employ the H.263 coding format whereas the wireless video-phone architectures have adopted the MPEG-4 coding format as well as H.263. Thus, this section looks further into this particular issue by discussing the major reason for the need of using different video compression standards. Moreover, possible real-life scenarios are also presented whereby an MPEG-4/H.263 heterogeneous video transcoder is found to be suitable.

4.3.1 The Importance of Using Two Different Standards

The MPEG-4 visual standard has unique features which were not addressed in the former video compression standards. These include object-oriented segmentation-based coding and robustness in mobile and wireless environments. Thus, MPEG-4 has primarily been introduced to satellite and mobile wireless multimedia communications. Meanwhile, H.263 will still be used in the circuit- and packet-switched networks for a long time to come.
While the current core H.263 standard appears to satisfy a number of existing applications, it does not support the eight key functionalities provided by the MPEG-4, such as content-based coding and superior channel error robustness. Therefore, H.263+ and H.263++ have recently been introduced. It is widely accepted that the current H.263 compression algorithm performs well for target applications operating at bit rates between 20 and 64 kbit/s. However, the compression efficiency and the associated quality have been proved to deteriorate rapidly at lower bit rates, such as 8–10 kbit/s [Lee3]. In the mean time, due to the rapid growth of the mobile and satellite communications, access to audio and video multimedia information via wireless networks has become extremely important.

Since transmissions over mobile and satellite networks are much more error-prone than the ones using fixed links, such as PSTNs and ISDNs, it is undoubtedly important to consider the error resilience performance and the capability of the video communication standards. This implies the need for a useful and reliable operation of audio and video compression algorithms suitable for error-prone environments at low bit rates, such as the error resilience functionality provided by the MPEG-4 [Tallu2]. Therefore, an MPEG-4/H.263 heterogeneous video transcoder is an important interface for the interconnection of these two video compression standards, particularly at low transmission rates.

### 4.3.2 Tandem in Asymmetric Networks Scenarios

In real-life scenarios, only a few of the networks are symmetric in topology. However, they can also be quite asymmetric, such as a fixed and a mobile-wireless network infrastructure or a fixed and a satellite network combination. Figure 4.2 illustrates these types of heterogeneous networking scenarios. The network on the left-hand side comprises the mobile-wireless cellular system. In this network, mobile multimedia terminals communicate with the base station (BS) of the cell in which they are located. The BS is connected to a mobile switching centre (MSC) which multiplexes the traffic of the other BSs. Finally, the MSC provides the connection of numerous BSs to the PSTN/ISDN via a video gateway. The video gateway accommodates a video transcoder which relays the input video stream to the output using the necessary translations between the different standards. A typical MSC is responsible for connecting as many as 100 base stations to the PSTN/ISDN lines. Thus, the connection between the MSC and the PSTN/ISDN requires substantial capacity at any time instant [Rappa].

The second asymmetric network scenario consists of the PSTN/ISDN and the very small aperture terminal (VSAT) satellite communications displayed on the right-hand side of the same figure. The multimedia VSATs communicate with the Hub terminal via the communications satellite and the Hub is similarly connected to the PSTN/ISDN through a video gateway.
4.4 Types of Heterogeneous Tandem of MPEG-4/H.263

This section presents information on two major types of tandem operation that provide a solution for the MPEG-4 and H.263 interoperability problem. The first type is the conventional method, as also described in the previous chapter. The second type is the heterogeneous video transcoding algorithm that is proposed during this research.

4.4.1 Conventional Cascaded Fully Decoding/Re-Encoding

The conventional method fully decodes the incoming compressed video stream at the gateway back to the pixel domain. Thus, this operation has to be accomplished by using the methods of the video compression standard which the input bitstream is originally encoded with. Following this operation, the reconstructed video data has to be fully re-encoded with the required video coding standard. The fully decode and re-encode operations require:

- The complex frame re-ordering, motion re-estimation, motion re-compensation and DCT/IDCT processes.

Consequently, this kind of operation consists of:

- A significantly time and power consuming, high complexity operation in the pixel domain.
- Reduced quality reconstructed video data.
When lossy compression techniques are used, such as in the MPEG-4 and H.263 standards, re-encoding typically incurs additional quality losses in the coded bitstream even though the same compression algorithm and bit rate are used for the input and output bitstreams. These losses stem from the fact that the video being re-encoded differs from the original video used in the initial encoder due to the lossy quantisation process of the DCT coefficients. Quality loss due to an extra decode/re-encode cycle has been effectively presented in the forthcoming results obtained from the various transcoding simulations in Figures 4.9-14.

The video gateway, in Figure 4.2, is the opening to the fixed network at which the decoding and re-encoding actions take place between the two different low bit rate video coding standards: MPEG-4 and H.263. However, this whole process can be very time consuming when the conventional cascaded fully decoding and re-encoding method is exploited, as discussed before. This aspect is quite important, particularly for the satellite communications. Since the delay factor of the satellite link is already quite high due to its nature, the decoding and re-encoding processes of the incoming bitstream at the gateway have to be avoided. Not only the delay factor, but also the QoS is severely degraded during the cascaded decoding and re-encoding processes. Therefore, transcoding between two standards is an important issue to be investigated as it involves the mapping of video parameters from H.263 to MPEG-4 or vice versa without the need of going through the conventional way of decoding and re-encoding, as sketched in Figure 4.3.

![Figure 4.3 The conventional way (cascaded fully decoding/re-encoding) versus transcoding](image)

### 4.4.2 Bi-Directional Heterogeneous MPEG-4/H.263 Video Transcoding

Developing an application which effectively provides an interface between two standards has always been a challenge. Thus, when developing such an application which operates between two different standards, the different approaches to problems taken by each standard can make the interoperability issue quite difficult. A number of these differences also exist between MPEG-4 and H.263. These diversities between the two low bit rate video compression standards are outlined in Table 4.1.


<table>
<thead>
<tr>
<th>Inter-Standard Differences</th>
<th>H.263</th>
<th>MPEG-4</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>video frame format</strong></td>
<td>fixed rectangular only</td>
<td>arbitrary shaped or fixed rectangular</td>
</tr>
<tr>
<td><strong>orientation</strong></td>
<td>frame-oriented only</td>
<td>object- or frame-oriented</td>
</tr>
<tr>
<td><strong>bitstream syntax</strong></td>
<td>headers, MVs, DCTs (no shape information)</td>
<td>headers, shape indices, MVs, DCTs</td>
</tr>
<tr>
<td><strong>frame headers</strong></td>
<td>straightforward</td>
<td>more elaborate and complex</td>
</tr>
<tr>
<td><strong>motion vectors</strong></td>
<td>64 MV differences</td>
<td>65 MV differences</td>
</tr>
<tr>
<td><strong>DCT coefficients</strong></td>
<td>1 Huffman VLC table</td>
<td>2 Huffman VLC tables</td>
</tr>
<tr>
<td><strong>intra DC values</strong></td>
<td>quantisation method difference</td>
<td>quantisation method difference</td>
</tr>
<tr>
<td><strong>chrominance motion</strong></td>
<td>no rounding parameter</td>
<td>rounding parameter</td>
</tr>
<tr>
<td><strong>compensation</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Table 4.1 Inter-standard differences between H.263 and MPEG-4*

The transcoding between H.263 and MPEG-4 is not an easy and straightforward operation as also observed from the number of discrepancies between the two standards in Table 4.1. Mainly, this difficulty is due to the fact that H.263 is a picture-dependent standard whilst MPEG-4 is content-based VOP-dependent. There are three parameters for MPEG-4, such as MVs, DCT coefficients and shape indices. Conversely, H.263 standard has two main parameters only, MVs and DCT coefficients. Therefore, mapping of the shape indices in object-oriented MPEG-4 to frame-oriented H.263 in which the shape information does not exist, is the main issue.

However, there are three major object profiles in the MPEG-4 standard: simple profile, core profile and main profile, as discussed in Chapter 2. An object profile is a special set of tools and/or algorithms that fulfils a given set of requirements. Full-scale compatibility of MPEG-4 with H.263 baseline coding is guaranteed with the simple profile in which the shape coding is not accomplished for the encoding of the video objects. The main application of this profile is the video-telephony and video-conferencing where head and shoulders types of images are typical [Schaf]. In addition to that, the MPEG-4 encoding algorithm has the ability to switch options for the use of different functionalities supported. In case of switching off those many functionalities that are not provided by H.263, the encoder operates with a very close resemblance to the H.263 baseline algorithm. In this case, the MPEG-4 video syntax is very similar to the current H.263 syntax. Only very few changes are required to achieve the tandem of both standards without the need for any further decoding and re-encoding processes [M3065, M3550].

The heterogeneous transcoding involves the mapping of one encoded bitstream to another in order that the target host video standard is enabled to decode it, as depicted in Figure 4.4. This operation comprises the necessary video syntax conversions in the compressed domain. Thus, the conversion algorithm only requires:

- Video frame header adjustment between the two standards.
- Video data translation from one standard to the other.
- Necessary bitstream stuffing for synchronisation purposes of the two standards, as seen in Figure 4.4.

![Figure 4.4 Heterogeneous MPEG-4/H.263 video transcoding essentials](image)

The straightforward translation and mapping operation of the transcoding also renders the computational complexity even less than a single decode operation. Similarly, the complexity of the transcoding is much less than that of the encoding process because:

- the complex frame re-ordering, motion re-estimation, motion re-compensation and DCT/IDCT operations, which consume much of the processing time and the power, are avoided.

Moreover, the heterogeneous video transcoding is accomplished in fully compressed domain, enabling the real-time operation, as opposed to the conventional cascaded fully decoding/re-encoding method. Therefore, this type of tandem:

- Presents a low complexity fast operation in fully compressed domain.
- Provides a minimal quality loss in the reconstructed images.

Having identified the mismatches between the two video coding standards in Table 4.1, it is notable to look further into each one of them for a successful interoperability operation. Thus, the forthcoming sub-sections give brief information on each of the transcoding issues followed by the proposed bi-directional MPEG-4/H.263 transcoder algorithm.

### 4.4.2.1 Differences in Video Frame Headers

There are significant differences between the video frame headers of the MPEG-4 and the H.263 bitstream syntax. H.263 has more compact and brief headers whereas MPEG-4 inserts more
complex headers at the beginning of each video frame. This is due to the fact that MPEG-4 has a layered structure of video objects, as mentioned in an earlier chapter of the thesis. Thus, each object layer has its own characteristic header data. On the contrary, since H.263 has a fixed rectangular frame format, it only evaluates a minimum amount of header data which is necessary for a successful decoding at the receiver end.

The video frame header structures of both standards are shown in Figures 4.5 and 4.6, respectively. As noted from the figures, MPEG-4 reserves more bits in its bitstream syntax for the
header information as each layered structure has its own specific data for signalling to the decoder end. Moreover, the number of functionalities supported by the MPEG-4 standard reflects the number of distinct header areas retained for these optional features which H.263 does not support. Therefore, the number of bits is much greater for the header syntax of the MPEG-4 data than the H.263 data. Figure 4.5 presents the header syntax of the MPEG-4 video when the number of functionalities are minimised. In this way, a closer video coding core to the H.263 standard can be provided, as discussed in the former sub-section. Thus, interoperability between the two standards is accomplished with the minimum amount of mismatches.

4.4.2.2 Differences in MV Tables

In the MV table of the MPEG-4 standard, an extra MV index is found which does not exist in the H.263 MV table. This extra vector difference is indexed as the 65th codeword in the table. The codeword of this particular MV is 0000 0000 0010 0 and it represents the MV difference for +16 pixels within any MB for high motion activity. Conversely, the H.263 standard bears only 64 differential MV indices.

4.4.2.3 Differences in DCT Coefficient Tables

MPEG-4 has an additional Huffman VLC table for encoding and decoding of the DCT coefficients. This additional Huffman VLC table is not defined for the H.263 standard. Thus, the compression and decompression operations of the transform coefficients which belong to the intra luminance blocks use this particular additional table. The VLC table for the evaluation of the intra chrominance, inter luminance and inter chrominance blocks is the same table as it appears in the H.263 standard.

4.4.2.4 Differences in Intra DC Quantisation Methods

The quantisation of the DC coefficients of the intra MBs presents two diverse methods. The two methods are theoretically presented in the following equations. Here, Equation (4.1) represents the quantisation algorithm for the DC values of the MPEG-4 intra MBs

\[
DC\_coeff = \frac{dc\_value}{8}; \quad 1 \leq QP \leq 4
\]

\[
DC\_coeff = \frac{dc\_value}{(2 \times QP)}; \quad 5 \leq QP \leq 8
\]

\[
DC\_coeff = \frac{dc\_value}{(QP + 8)}; \quad 9 \leq QP \leq 24
\]

\[
DC\_coeff = \frac{dc\_value}{(2 \times QP - 16)}; \quad 25 \leq QP \leq 31
\]

whilst Equation (4.2) formulates the quantisation theory for the DC values of the H.263 intra MBs, respectively.

\[
DC\_coeff = \frac{dc\_value}{8}; \quad 1 \leq QP \leq 31
\]
In the above equations, $Q_P$ represents the quantisation parameter, $dc\_value$ is the DC coefficient of an intra MB after the DCT is taken and finally, $DC\_coeff$ stands for the DC coefficient when the quantisation process is applied to the $dc\_value$. The incorrect application of these two methods significantly perturbs the transcoded video frames. As the DC values of the intra MBs predominantly affect the rest of the transcoded blocks and frames, the quantisation level differences result in luminance and chrominance data deterioration. It is a very annoying occurrence for the human eye as the luminance is greatly damaged and the colour data is dominated by either a pinkish or a greenish effect.

4.4.2.5 Differences in Motion Compensation with Rounding Parameter

Finally, in the MPEG-4 video coding standard, a 1-bit field is signalled to the decoder through the video frame headers of each VOP comprising an attribute which is not shared with the H.263 standard. This particular 1-bit flag signals the value of the parameter rounding control used for the pixel value interpolation in motion compensation for the inter (P) and sprite (S) VOPs, or in other terms video frames. This control bit does not exist for the intra (I) VOPs. When this flag is set to 0, the value of the rounding control is 0 and when it is set to 1, the value of the rounding control is also 1. When the rounding parameter is not present in the VOP header of the MPEG-4 video frame, the value of the rounding control is accepted as 0. Moreover, the value of the rounding parameter alternates between 0 and 1 with the successive video frames.

The rounding control is effective during the half pixel interpolation within the motion compensation process, as stated earlier. Half pixel values are calculated using the bi-linear interpolation, as described in Figure 4.7, in both of the standards. However, the methods show the discussed discrepancies, as shown in the following mathematical equations.

![Figure 4.7 Bi-linear interpolation scheme for half sample search for both standards](image)

**LEGEND:**

- $\times$ Integer pixel position
- $\circ$ Half pixel position

Equation (4.3) presents the interpolation method used in the MPEG-4 standard.
whilst Equation (4.4) demonstrates the interpolation method used in the H.263 standard.

\[
a = A \\
b = (A + B + 1 - \text{rounding}_\text{control}) / 2 \\
c = (A + C + 1 - \text{rounding}_\text{control}) / 2 \\
d = (A + B + C + D + 2 - \text{rounding}_\text{control}) / 4
\]

4.4.2.6 Bi-Directional Heterogeneous MPEG-4/H.263 Video Transcoder Algorithm

Figure 4.8 illustrates a more detailed operation of the proposed MPEG-4/H.263 video transcoding algorithm whereby the input bitstream is simultaneously mapped onto the output bitstream. Here, the mapping process starts with the extraction of the headers in the bitstream which belong to the preceding host video standard and insertion of the new ones which are decodable by the next standard. The data which is common for both is maintained at the transcoder. Particularly, the mapping of the headers from H.263 to MPEG-4 needs careful attention as MPEG-4 bears elaborate header information as opposed to H.263.

Once the header translation has been completed for a video frame, time comes to map the data of that particular frame. The orders of the bitstream syntax data for both standards and the syntax mapping process are shown in Figure 4.8. As it is observed from the figure, having handled the
frame headers, the transcoder translates and maps the contents of the MB headers, the MVs and the DCT coefficients without any decoding/re-encoding operations.

This particular stage comprises the most important and demanding part of the whole transcoding process. Even though both standards are in common from many aspects and in close resemblance to each other from the algorithmic point of view, there are still certain discrepancies that should not be ignored. For instance, the two standards have a noticeable difference in the methods they use to quantise and de-quantise the DC coefficients of the intra coded MBs and blocks. Therefore, at the mapping stage of the two bitstreams, the incoming DC values should be de-quantised and re-quantised by the methods provided by the posterior video standard. Since the transcoder does not involve any bit rate conversions here, re-quantisation of the DC values does not cause any quality degradation in the transcoded picture due to the drift problems as in the homogeneous video transcoding. The quantisation parameter is kept constant during the video standard conversions. In this way, the motion estimation and compensation of the predicted frames are carried out without any mismatches.

Another major distinction arises between MPEG-4 and H.263 is the existence of a second Huffman VLC table for the encoding and decoding of the DCT coefficients on the MPEG-4 side, which is not defined in the H.263 algorithm. The transform coefficients of the intra luminance blocks in MPEG-4 are encoded and decoded by the use of this second VLC table whilst all the H.263 blocks use only one table, which is also common in both standards. For an achievable interoperability between the two standards, this distinction has to be taken into consideration during the transcoding process. This can be accomplished by merely re-evaluating these blocks during the transcoding from MPEG-4 to H.263. Clearly, this problem does not occur in the reverse direction.

Moreover, the MPEG-4 MV table has one extra vector index that helps the picture quality improve in high motion areas. Therefore, in case of translation from the MPEG-4 syntax to H.263, this extra vector index should be mapped to the nearest vector difference, so that the H.263 decoder is enabled to decode it from its built-in table. In the opposite direction, the H.263 MV indices are also contained in the MPEG-4 table and hence, no further action is required.

One last difference between the standards occurs in the motion compensation stage of the predicted frames. The MPEG-4 standard uses a rounding parameter which does not exist in H.263. This parameter has arbitrary values during the motion compensation of the predicted MBs, and it is signalled in the MPEG-4 header data [MPEG4, N1796]. For a successful pixel matching basis, this parameter should be forced to have a null value in order to comply with its absence in H.263. The ignorance of the existing rounding parameter whilst transcoding results in a loss of colour information. Since both MPEG-4 and H.263 exploit the compression capabilities of the motion-
compensated predictive coding, the colour deterioration propagates throughout the entire transcoding process resulting in a very annoying effect on the viewer.

During the simulations, the effect of changing this rounding parameter to zero was experienced as a negligible loss on the PSNR levels of the MPEG-4 video. Moreover, the recent versions of the MPEG-4 software have been designed in such a way that the rounding parameter can be controlled by the user externally. Thus, there is no need for an internal zero value enforcement.

4.5 Computer Simulations and Analysis of the Results

In this section, the computer simulation results are given along with their discussions. The simulation conditions are further detailed within the section. The results are presented in a comparative way which clearly identifies the pros and cons of the two types of tandem discussed in the former section. Both the objective and subjective simulation results are demonstrated in this section.

4.5.1 Experiments and Results

The heterogeneous transcoding simulations were carried out by using three sets of video test sequences: 150-frame “Suzie” sequence, 200-frame “Foreman” and “Carphone” sequences. The choice of test sequences reflects the need for performance evaluation of the proposed and designed video transcoding algorithm under varying motion activity conditions and hence, from moderate to very high motion activity video test clips are used: “Suzie”, “Carphone” and “Foreman”, respectively. As the amount of motion activity increases within a video sequence, the number of intra coded MBs and/or four MV sets per one MB cases proportionally increases. Naturally, the consequence of this incident is the increase in the average output bit rate for the particular high motion areas.

The results are obtained and presented for both directions, from MPEG-4 to H.263 and vice versa, and also for three different simulations: (1) direct encoding/decoding, (2) transcoding and (3) cascaded decoding/re-encoding. The direct encoding/decoding results are considered as the reference values for the simulations. Figures 4.9, 4.11 and 4.13 demonstrate the objective results and Figures 4.10, 4.12 and 4.14 depict the subjective results for the simulations of the three test sequences, respectively. All sequences were encoded in I-P-P-P-P-... format at 25 f/s. The quantisation parameters were kept constant at 10 without any rate controlling algorithm in each of the simulations. Therefore, as seen in the figures, the transcoded sequences in both directions do not appear to have any drift effect which deteriorates the picture quality.
The transcoder performance evaluation was carried out by running the MPEG-4 codec in the baseline mode without the shape coding features [Schaf]. In the H.263 to MPEG-4 direction, the compatibility was achieved without the use of negotiable options. In the reverse direction, however, advanced prediction and unrestricted modes of the MVs were set, giving rise to an increase of around 0.5 dB in the PSNR levels due to the use of four MVs per one MB. This effect can be observed by comparing the left-hand side graphs of Figures 4.9, 4.11 and 4.13 with the right-hand side graphs.

Figure 4.9 PSNR variations with and without transcoding for the 150-frame “Suzie” sequence at 56 kbit/s on average with a frame rate of 25 fr/s

Figure 4.10 Subjective results of the 150th frame of the “Suzie” (56 kbit/s on av.) sequence. Top row- MPEG-4 to H.263 path; bottom row- H.263 to MPEG-4 path. First column- direct encoding/decoding; second column- transcoding; third column- cascaded decoding/re-encoding
Chapter 4  
Heterogeneous Video Transcoding

MPEG-4 -> H.263 route

<table>
<thead>
<tr>
<th>Frame Number</th>
<th>PSNR (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>31</td>
</tr>
<tr>
<td>50</td>
<td>33</td>
</tr>
<tr>
<td>100</td>
<td>34</td>
</tr>
<tr>
<td>150</td>
<td>33</td>
</tr>
<tr>
<td>200</td>
<td>32</td>
</tr>
</tbody>
</table>

H.263 -> MPEG-4 route

<table>
<thead>
<tr>
<th>Frame Number</th>
<th>PSNR (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>31</td>
</tr>
<tr>
<td>50</td>
<td>33</td>
</tr>
<tr>
<td>100</td>
<td>34</td>
</tr>
<tr>
<td>150</td>
<td>33</td>
</tr>
<tr>
<td>200</td>
<td>32</td>
</tr>
</tbody>
</table>

**Figure 4.11** PSNR variations with and without transcoding for the 200-frame “Carphone” sequence at 82 kbit/s on average with a frame rate of 25 fr/s

![PSNR variations graph](image)

**Figure 4.12** Subjective results of the 200th frame of the “Carphone” (82 kbit/s on av.) sequence.

*Top row-* MPEG-4 to H.263 path; *bottom row-* H.263 to MPEG-4 path. *First column-* direct encoding/decoding; *second column-* transcoding; *third column-* cascaded decoding/re-encoding
Figure 4.13 PSNR variations with and without transcoding for the 200-frame "Foreman" sequence at 100 kbit/s on average with a frame rate of 25 fr/s.

Figure 4.14 Subjective results of the 200th frame of the "Foreman" (100 kbit/s on av.) sequence. 
Top row- MPEG-4 to H.263 path; bottom row- H.263 to MPEG-4 path. First column- direct encoding/decoding; second column- transcoding; third column- cascaded decoding/re-encoding.
4.5.2 Analysis of the Results

The left-hand side columns of Figures 4.9, 4.11 and 4.13 illustrate the PSNR variations for the MPEG-4 to H.263 direction whilst the right-hand side columns of the same figures present the reverse direction PSNRs. In the H.263 to MPEG-4 paths, the PSNRs of the transcoded frames are only on average 0.05 dB less than the reference values whilst the ones in the reverse paths are almost similar to those of the reference frames. Thus, all six graphs present that the transcoding results give a performance at least as good as the direct encoding/decoding simulation results. Conversely, the cascaded decoding/re-encoding PSNRs are on average 1–1.5 dB less than the transcoding PSNRs. This is due to the reason that the transcoding exploited the MVs of the incoming bitstreams without decoding them whereas in the cascaded decoding and re-encoding operations, new MVs were calculated based on the lossy reconstructed pictures. Furthermore, the transcoded video data did not suffer from the lossy de-quantisation and re-quantisation effects as opposed to the cascaded method. As discussed above, an average of 0.5 dB difference between the two sets of results was caused by the usage of four MVs per MB for MPEG-4 to H.263 direction which increased the overall picture quality, particularly in high motion areas. In the reverse direction, the similar consequence cannot be observed as the H.263 to MPEG-4 path did not enable the four MVs per one MB feature.

The subjective results of the 150th frame of the “Suzie”, the 200th frames of the “Carphone” and “Foreman” sequences can also be seen in Figures 4.10, 4.12 and 4.14, respectively. The top rows of each set depict the MPEG-4 to H.263 direction whilst the bottom rows show the H.263 to MPEG-4 direction for the direct encoding/decoding, transcoding and cascaded decoding/re-encoding results, respectively. The third columns of each set of figures present some degraded quality in comparison with the others due to the distortion caused by the excessive quantisation losses and the re-evaluation of the previously estimated and compensated MVs of the decoding and re-encoding processes. The distortion is much clearer for the “Foreman” sequence compared to the other presented sequences as this particular sequence is known and accepted to be a highly active test sequence comprising significant amount of motion.

Furthermore, the transcoding had significantly lower complexity since it employed a straightforward syntax and bit pattern mapping process without the need for the very complex motion estimation, compensation and DCT/IDCT processes.

The experiments were also repeated for other video test sequences and very similar conclusions were derived. The reduction in the picture quality was seen to be subjectively imperceptible with the exploitation of the transcoding methods.
4.6 Concluding Remarks

The communications evolution towards the 3G and universal mobile telecommunication service (UMTS) [Dahlm] systems has increasingly demanded the provision of the most efficient and optimised solutions to the interoperability issues. The problem of inter-working of numerous diverse networks has been addressed many times in the referred communication systems. Since the convergence of various heterogeneous multimedia networks has been a key topic for the design paradigm for such international telecommunication standards, a study in this direction has also been carried out in this research.

Therefore, this chapter has been assigned for the discussion of such interconnectivity architecture. This architecture has been built up based on the major two low bit rate video coding standards, namely H.263 and MPEG-4. These two standards are believed to play a primary role on the low bit rate video communications for the two-way video-conferencing/telephony applications as well as multimedia data remote log-in/access operations.

In this chapter, a bi-directional MPEG-4/H.263 heterogeneous video transcoding algorithm has been presented. This has entirely been a novel proposal of a video transcoder design for the efficient interconnection achievement of the two low bit rate video compression standards in the transcoding literature. This study has produced two papers, [Dogan2] and [Dogan4], to set new milestones in the heterogeneous video transcoding research. The positive effects of the two papers have been realised as a trigger for more recent follow-on publications, such as [Feams, Iwasa, Shanah-1-3]. Thus, the novelty that has been presented in this piece of research work can briefly be summarised by the following bullet-points:

* The optimisation of the inefficient operation of a cascaded fully decoding and re-encoding processes for the MPEG-4 and H.263 standards leading to:
* The design of a novel efficient MPEG-4/H.263 video transcoding algorithm which fully interconnects the two low bit rate video coding standards in the compressed domain bi-directionally.

The success of such architecture comes from its low complexity and improved quality operation. The following table, Table 4.2, summarises the pros and cons of the two schemes, namely the conventional and the proposed transcoding methods, comparatively.

<table>
<thead>
<tr>
<th>Key Features</th>
<th>Conventional</th>
<th>Heterogeneous Transcoding</th>
</tr>
</thead>
<tbody>
<tr>
<td>domain</td>
<td>fully pixel</td>
<td>fully compressed</td>
</tr>
<tr>
<td>required operations</td>
<td>fully cascaded decode/re-encode</td>
<td>on-line syntax translation &amp; mapping</td>
</tr>
</tbody>
</table>

112
### Heterogeneous Video Transcoding

#### Chapter 4

**applied algorithms**
- frame re-ordering, motion re-estimation, re-compensation, inverse Q/re-Q, IDCT/DCT

**complexity**
- very high due to an extra decode/re-encode cycle
- even lower than one decode operation

**video quality**
- 1-1.5 dB quality loss due to lossy de-quant/re-quant operations & low quality motion re-evaluation
- as good as direct encode/decode process with minimal quality degradation

**real-time operation**
- incurs significant processing delay due to very high algorithmic complexity
- suitable for such implementation due to low complexity low delay operation

<table>
<thead>
<tr>
<th>applied algorithms</th>
<th>complexity</th>
<th>video quality</th>
<th>real-time operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>frame re-ordering, motion re-estimation, re-compensation, inverse Q/re-Q, IDCT/DCT</td>
<td>very high due to an extra decode/re-encode cycle</td>
<td>1-1.5 dB quality loss due to lossy de-quant/re-quant operations &amp; low quality motion re-evaluation</td>
<td>incurs significant processing delay due to very high algorithmic complexity</td>
</tr>
<tr>
<td>frame header extraction/insertion, MVs &amp; DCT coefficients mapping</td>
<td>even lower than one decode operation</td>
<td>as good as direct encode/decode process with minimal quality degradation</td>
<td>suitable for such implementation due to low complexity low delay operation</td>
</tr>
</tbody>
</table>

**Table 4.2** The distinctive features of the two tandem methods for a quick comparison
Chapter 5

5 Error-Resilient Video Transcoding

5.1 Introduction

Raw video data bears a fairly high proportion of redundant information compared to the essential data within a video stream. Compression of the video stream is achieved through the removal of this redundancy from the rest of the video data. This process renders the video material very sensitive to transmission errors over wireless channels whilst providing substantial amount of bit rate reduction. Therefore, the challenge has always been to protect the video communications with various error protection schemes and resilience algorithms prior to transmission from the source encoder. Depending upon the video communication applications, the protection or resilience is accomplished at the cost of increased transmission rates [Dogan1, Tallu2].

Nevertheless, as opposed to the conventional source-driven resilient transmissions, recent research is focusing on the addition of resilience to the video data where or whenever it is needed. Bearing this in mind, error resilience can also be introduced into an already encoded video stream at an intermediate stage. This particular stage where the addition of error resilience to the video stream takes place can simply be the video gateway within the entire network [Reyes1, Reyes2]. The video gateway comprises a video transcoder or a set of transcoders that provides the necessary bit rate management between different networks. Therefore, problems arising from bandwidth limitations can be resolved dynamically within the network without involving the source of the communication. This evidently enables faster system responses and more efficient congestion control techniques with the utilisation of the useful features of the video transcoders [Dogam3].

The error resilience is applied to the encoded video stream with the aim of providing robustness against transmission errors. Despite the fact that this process is conventionally carried out at the source encoder, [Reyes1] describes how resilience can also be inserted into the compressed video bitstream by means of a video transcoder at a centralised point within the network. On the other hand, a video gateway that contains a video transcoder is capable of adding robustness to the already compressed video stream both when and wherever it is required. Figure 5.1 displays a typical scenario where a video transcoder is required to provide resilience between two separate networks. The gateway interconnects a relatively low BER and high bandwidth network, such as
the ISDN and/or PSTN shown to the left side of the figure, to a relatively high BER and low bandwidth network, like the mobile-wireless network displayed on the right-hand side.

In addition, the video gateway can also employ error concealment prior to the transcoding process, in order to reduce the destructive effects of transmission errors. Figure 5.2 demonstrates such a case. This particular figure also highlights the necessity of the interoperability between two or more high BER networks, such as a mobile-wireless network or a high packet-loss-ratio (PLR) network, like the Internet. The gateway can apply error concealment on the partially decoded video data using one of the schemes discussed in [Wang]. The output bit rate from the gateway can be adjusted by monitoring the occupancy of frame buffers situated at the end of the video transcoding block. The state of these buffers varies according to the channel bandwidth conditions. The amount of resilience added to the video data can also be controlled by monitoring the transcoder output rate and the changing error conditions of the network. This is accomplished by the means of feedback signalling, as shown in Figures 5.1 and 5.2.

![Figure 5.1 Error-resilient video transcoding scenario](image)

The video gateway can be designed in such a way as to apply error resilience by partitioning the data and inserting re-synchronisation markers into the pre-encoded non-resilient bitstream during the actual transcoding process. In addition to data partitioning, the transcoder can apply unequal error protection to different parts of the video data. Therefore, the most significant data, namely the header data, is assigned the highest protection. The MVs and DCTs are also protected during transcoding, each with a decreasing level of protection, respectively. Furthermore, the video gateway can select different channels for transmitting various parts of the video stream, depending on the error characteristics of each of the channels and the error sensitivity of video parameters. Such a process enables the video gateway to facilitate multi-threading [Wenge1, Wenge2], so that the important portions of the video data are prioritised and protected according to their significance. A similar procedure is also employed in some systems, such as the general packet
radio service (GPRS) and the enhanced data rates for global system for mobile telecommunication (GSM) evolution (EDGE) [Furus], where the encoded data, which has been separated into different portions, is allocated different levels of protection.

![Error resilience control feedback](image)

**Figure 5.2 Error-resilient video transcoding with error concealment scenario**

Moreover, the AIR algorithm [MPEG4, N1646] can be applied to the transcoded video frames in order to prevent the temporal error propagation throughout the video sequence [Dogan5, Reyes2]. However, this scheme tends to increase the output bit rate. In order to compensate for this, the rate management feature of the video transcoding algorithm is employed, so that the video gateway produces a resilient video stream with a controlled output rate [Dogan5]. [Reyes2] also describes how the error resilience operation of a video transcoder can be exploited so as to provide additional spatial error protection during the transcoding operation. In this particular method, regular re-synchronisation words are inserted into the input compressed video stream to maintain spatial synchronisation. In addition, the video transcoder can be engineered to adapt its transcoding method and parameters based on feedback signals from the end-users to counteract the quality degrading effects of the frame losses during transmissions. The adaptive transcoding can thus provide a protection against temporal error propagation which in turn results in a reduction in the deterioration of the output quality caused by the frame losses. This particular process may require a multi-frame buffer to store the previously transcoded frames, which can provide a better reference when frames are lost during transmission [N1646]. Furthermore, [Swann] shows how the error-resilient entropy coding (EREC) algorithm [Redmi] can also be exploited as a video transcoding mechanism to provide error resilience.

Various other resilience techniques [N1646] can also be applied during transcoding. The selection of the most suitable error resilience scheme is based on the quality improvement features that it provides and also its complexity. In addition, the network characteristics also need to be
considered. By moving the error resilience support from the source encoder to the video gateway, a more rapid and dynamic way of error-handling within the network is achieved.

Error resilience was described in detail in Chapter 2. This chapter focuses more on the combination of the particular two resilience schemes, which were also discussed previously, whilst preserving the transmission rate management features of the video transcoders. Thus, this chapter is organised as follows: The second section discusses the different communication scenarios where the error-resilient video transcoding can possibly be required. The third section describes the resilient video transcoding architecture and the fourth section demonstrates the experiments and computer simulation results carried out using the discussed scenarios. This section comprises the tests of two major error resilience techniques, namely the AIR and FCS methods, applied to the rate management scheme. Finally, the fifth section concludes the chapter.

5.2 Error-Resilient Video Transcoding Requirement Scenarios

Regulation of the bit rate from a higher to a lower degree has become a requirement due to the wide range of multimedia applications and their users with different needs and expectations. Today, accessing the pre-encoded video data within the video source is not a significant problem through wired or wireless means. Moreover, two-way video communications and video-conferencing have already been introduced both in fixed and mobile networks. However, user constraints, such as the provision of a reduced bandwidth and a limited source power support, impose a harsh limit on the QoS of these types of video transmissions. Furthermore, the video quality is particularly perturbed with transmission errors within the communication channels, such as in wireless video transmissions. This is due to the fact that the wireless video communications are known to be vulnerable to destructive channel errors. Thus, a video transcoder that translates higher bit rates to lower ones will also be expected to produce error-resilient outputs in the near future. Therefore, this chapter proposes and describes an error-resilient MPEG-4 video transcoding algorithm for bit rate management.

The two problems addressed above, namely the bit rate regulation and error resilience, are of the main concerns of this chapter. Thus, two main functionalities have been incorporated within the video transcoder for an achievable interconnection between the heterogeneous types of multimedia networks. A possible real-life scenario can be seen in Figure 5.3, which depicts two different networks, one of which has congestion problems at a given time (network-3) and another with bandwidth constraints due to its wireless nature (network-2). The two problems impose an
action to be taken by the video gateway in order to provide the clients of such networks with a satisfactory service at the required transmission rates. Both networks on the right-hand side of the video gateway cannot tolerate equivalent bit rates to the networks shown on the left. In such cases, the video stream suffers from a continuous quality degradation at the video gateway whilst crossing from network-1 to network-2/3. One cruel method to stop this happening is to tell the video source to reduce its bit rate by means of a feedback channel. This can be accomplished by monitoring the network that is most suffering from the congestion problems. However, this solution will definitely not be the best possible approach since there is the fourth network (network-4) in the whole scenario without any channel bandwidth constraints, which actually can accommodate bit rates matching the source data rates. The fair solution to this problem is employing a video transcoder at the video gateway which will regulate the rate of the video stream according to the needs of individual networks and their bandwidth characteristics [Dogan3, Warab].

![Figure 5.3 A heterogeneous multimedia networking scenario using a video transcoder at the video gateway](image)

The bit rate management at the gateway solves the problem without the need for a complex source with long feedback delay times and quality degradations, as also elaborated in Chapter 3. However, there still exists a problem for the wireless network, network-2, which is the robustness to the wireless channel conditions. To tackle this particular problem, an error resilience scheme is required for the operation of the video transcoder to give a better service quality over error-prone channels. Therefore, the video transcoder has to be tailored in a resilient fashion to employ spatial and/or temporal resilience. In this way, the error propagation can be avoided whilst achieving the
necessary multimedia traffic planning through the error-resilient video transcoding. Besides, the standards-compliant video output from the video transcoder has to be preserved.

![GPRS networking scenario with an error-resilient video gateway](image)

**Figure 5.4** A GPRS networking scenario with an error-resilient video gateway

A similar need for the error-resilient handling of the transcoded video stream may arise over mobile-access networks, such as GPRS. This comprises a typical networking problem of the 3G communication scenarios. The nature of the GPRS channels is characterised by bursty errors causing deep fades of the signal strength caused mainly by the co-channel interference and the multipath effects. Due to this fact, the video transmission will greatly be affected over the GPRS channels resulting in perturbed images with significantly reduced QoS levels. Thus, during the access via GPRS, video gateways will play an important role not only matching the transmission rates to the user requirements, but also providing the necessary protection for the transcoded video streams prior to their transmissions, as depicted in Figure 5.4. The following sub-section is dedicated for the brief overview of the GPRS networks that will later be used for the experimentation of the error-resilient video transcoding model within this chapter.

### 5.2.1 General Packet Radio Service (GPRS) Networks

GPRS [GSM0364] is a new non-voice value added service that allows information to be sent and received across a mobile telephone network. It is an end-to-end mobile packet communication system which makes use of the same radio architecture as GSM [Brasc, GSM0364]. GPRS is also the name for an international packet-switched networking standard in GSM systems, initiated and developed by the European Telecommunication Standards Institute (ETSI).

GPRS offers theoretical maximum air-interface transfer rates of up to 171.2 kbit/s with its multi-slotting capability. This is significantly faster than the data transmission speeds possible over
today's fixed telecommunication networks and the current circuit-switched data services on GSM networks. Thus, GPRS promises to fully enable the use of new applications on the move with the increased communication speeds. However, achieving the theoretical maximum GPRS data transmission speed requires that a single user takes over all the dedicated timeslots without any error protection. Clearly, since it is unlikely that a network operator will allow all timeslots to be used by a single GPRS user, transfer rates lower than 171.2 kbit/s are more likely to be pronounced in the realistic systems subject to the mobile terminal capabilities and carrier interference. Moreover, GPRS facilitates instant connections whereby information can be sent or received immediately as the need arises, subject to the radio coverage. Consequently, with the use of GPRS, there is no need for a dial-up modem connection [Cat, Rysav].

GPRS involves overlaying a packet-based air interface on the existing circuit-switched GSM network. This gives the user an option to use a packet-oriented data service. Therefore, with the use of GPRS, the information is split into separate but related packets before being transmitted and re-assembled at the receiving end. User data packets are segmented, coded and transformed into radio blocks. Each radio block is further interleaved over four standard GSM normal bursts and transported across the air interface in the same manner as the circuit-switched speech is transmitted in GSM. When error occurs in the transmission medium, data packets can be re-transmitted at the radio block level. The set of bursts that results from a single user data packet is marked with a temporary flow identifier (TFI) which is then used on the receiving side to re-assemble the user data packet [Betts, Granb].

<table>
<thead>
<tr>
<th>Coding Scheme</th>
<th>Convolutional Code Rate</th>
<th>Payload per Block [bits]</th>
<th>User Bit Rate [kbit/s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS1</td>
<td>1/2</td>
<td>181</td>
<td>9.05</td>
</tr>
<tr>
<td>CS2</td>
<td>~2/3</td>
<td>268</td>
<td>13.4</td>
</tr>
<tr>
<td>CS3</td>
<td>~3/4</td>
<td>312</td>
<td>15.6</td>
</tr>
<tr>
<td>CS4</td>
<td>1</td>
<td>428</td>
<td>21.4</td>
</tr>
</tbody>
</table>

Table 5.1 GPRS channel coding schemes

A new set of logical channels has been defined for GPRS traffic as opposed to the circuit-switched networks where all the signalling and information transfers make use of one channel only. This set includes control channels and packet data traffic channels. A physical channel allocated for GPRS traffic is called a packet data channel (PDCH). The PDCH consists of a multi-frame pattern that runs on timeslots assigned to GPRS [Betts, Cai]. Thus, the GPRS data is transmitted over the PDCH and is protected by four different channel protection schemes: CS1, CS2, CS3 and CS4 [GSM0503]. The channel coding is used to protect the transmitted data packets against transmission errors. CS1-3 use convolutional codes and block check sequences of varying strengths so as to produce different rates. CS1-3 are based on a 1/2 rate convolutional code, which is punctured to obtain approximate rates 1/2, 2/3 and 3/4, respectively. On the other hand, CS4 is
uncoded whereby it only provides error detection functionality [Larss]. Each of the four channel protection schemes is assigned a maximum of eight timeslots [GSM0364]. The coding schemes and resulting bit rates per one timeslot are described in Table 5.1.

The choice of one of the four coding schemes for the coding of PDCHs depends on the quality of the channel and also on the application's QoS requirements. Under very bad conditions, a very reliable CS1 may be used and a data rate of 9.05 kbit/s per GPRS timeslot can be obtained. Under good channel conditions, data can be transmitted without convolutional coding and a transport rate of 21.4 kbit/s per timeslot can be achieved. Consequently, with the use of eight slots of this channel coding scheme, namely CS4, a maximum data rate of 171.2 kbit/s can be obtained in theory. However, in practice, multiple users may sometimes share the timeslots resulting in a much lower bit rate for an individual user [Granb].

Furthermore, several mobile terminals can dynamically share the pool of PDCHs in one cell and several PDCHs can also be used simultaneously for a single connection. Such a scenario is displayed in Figure 5.5, where the GPRS resources are shared among multiple users. Thus, a user data packet can be transmitted over multiple PDCHs and re-assembled at the receiving end, as illustrated in Figure 5.5. During the transmission of the user data packets, the network controls the allocation of resources by necessary signalling and the receiving terminal identifies its packets via the TFI.

Packet switching means that GPRS radio resources are used only when users are actually sending or receiving data which results in remarkable spectrum efficiency. Rather than dedicating a radio channel to a mobile data user for a fixed period of time, as in the circuit-switched GSM systems,
the available radio resource can be concurrently shared between several users with the GPRS networking. This efficient use of the scarce radio resources means that large numbers of GPRS users can potentially share the same bandwidth and be served from a single cell. However, the actual number of supported users highly depends on the application being used and the amount of data being transferred. Thus, GPRS offers a billing scheme which is relative to the amount of data transmission, as opposed to circuit-switched systems (i.e. the conventional GSM) whereby billing is mainly based on the duration of the connection [Chiar].

From Internet web browsing to e-mailing/chatting, a wide range of corporate and consumer applications can be enabled by non-voice mobile services which are offered by GPRS [Bucki, Rysav]. Amongst those many applications, two of the particular GPRS features offer a suitable medium for the long-awaited image and video communications. These two features are the multi-slotting capability of GPRS, which effectively allows for the increased throughputs for a single mobile unit, and the interworking with Internet multimedia applications with its native Internet protocol (IP) support. Over time, the nature and form of mobile communications are getting less textual and more visual. The wireless industry is moving from text messages to icons, picture messages to photographs, blueprints to video messages and movie previews being downloaded to entire movie clips via data streaming on a mobile device.

Sending moving images in a mobile environment has several market applications, including monitoring parking lots or building sites against the intruders or thieves and sending images of patients from an ambulance to a hospital or from an emergency scene to the relevant authorised headquarters. Moreover, video-conferencing applications are another rapidly emerging type of applications for moving images which will exploit the features offered by the GPRS technology.

### 5.3 Error-Resilient Video Transcoder Architecture

In this chapter, the video transcoding has further been exploited to add error resilience to the transcoded data in addition to the rate management characteristics. For this purpose, the transcoding system has been modified, as illustrated in Figure 5.6. Referring to this figure, the video transcoder reduces the incoming bit rate whilst adding resilience to the transcoded video data simultaneously. The rate reduction algorithm provides drift-free transcoding qualities with refined MVs. Furthermore, the increase in the output bit rate due to the addition of resilience is compensated with an adaptive transcoding operation. The resilience is provided with the use of AIR and FCS algorithms, the details of which were discussed in Chapter 2.
Adaptive transcoding imposes a feedback signalling scheme for the control of the output bit rate from the video transcoder. The feedback signal is originated from the output video frame buffer which constantly monitors the flow conditions. In the case of an underflow, it returns a signal to the transcoder to initiate an increase in the output rate. On the other hand, the rate reduction is flagged back to the transcoder in the case of an overflow. Thus, a straightforward rate controlling scheme is established for a congestion control or a bandwidth bottleneck resolution. However, this feedback signalling is not only used for the straight rate controlling algorithm, but also for the amount of resilience to be added to a particular video frame prior to its transmission. Therefore, the back channel signal carries an extra useful data which bears the information of the varying error-prone channel conditions.

Since both the AIR and FCS methods increase the overall transmission rate, the video transcoder adaptively transforms the bit rate to the rate required by the congested or bandwidth-limited network(s), as explained in Chapter 3. The rate regulation is simply carried out by the adaptation of the QP to the newly required conditions. During transcoding, an increase in QP results in a bit rate reduction whilst a decrease gives an increase in the transcoder output rates. The operation of this rate management particularly for AIR is as follows:

- **Step-1.** Start transcoding with the pre-determined number of I-MBs for AIR.
- **Step-2.** Check whether the target bit rate was previously met.
  - **Step-2.a.** If not, check whether the mobility is higher in this particular frame compared to the former one.
    - **Step-2.a.1.** If so, transcode the frame with one additional number of I-MB.
    - **Step-2.a.2.** If not, transcode the frame with the same number of I-MBs as before.
  - **Step-2.b.** If so, check whether the mobility is higher in this particular frame compared to the former one.
    - **Step-2.b.1.** If so, transcode the frame with the same number of I-MBs as before.
Step-2.b.2. If not, transcode the frame with one less I-MBs.

- Step-3. Check whether the target bit rate is now met.
  - Step-3.a. If not, keep the QP as it is or decrease it by one depending on the feedback signal showing the status of the output frame buffer underflow. Signal it to the resilient coding part.
  - Step-3.b. If so, increase the QP by one depending on the feedback signal showing the state of the output frame buffer overflow. Signal it to the resilient coding part.

- Step-4. Repeat steps from 2 to 4.

This particular operation of the video transcoder regulates the output bit rate whilst also introducing the best possible amount of resilience to the video stream for the high motion areas with the use of AIR. Thus, in addition to processing the high motion data in an error-resilient way, the transcoder also encodes these particular portions of the video sequence with an increased number of I-MBs whilst compensating for the resulting increased bit rates. Thus, it can be noted that an error-resilient video transcoder makes a further use of its rate regulatory feature to employ more efficient resilience schemes without any consequential growth in its output rate.

Furthermore, the video transcoder is designed in a fashion that makes it capable of receiving any kind of transmission feedback signal, such as ACK, NACK and/or both, from the end-receiver. Depending on the received return signal from the end-user, the video transcoder adapts its transcoding scheme according to the error-prone channel conditions. This is a similar transcoder operation to the operation of a source coder described in Chapter 2 for the FCS algorithm. However, this kind of an error-resilient video transcoder is also capable of regulating the increased transmission rate due to the FCS algorithm here. According to the feedback signal obtained from the receiving end, the video transcoder can judge which video frames are not correctly received and/or lost during transmission. As a result, error resilience is applied to the subsequent transcoded frames by using the previously stored video frames in the transcoder buffer. Thus, a certain degree of error resilience is inserted by referring to the error-free video frames from the transcoder buffer; hence, resulting in a higher QoS provision. In this way, the error propagation effects can be minimised at a much earlier point within the entire network rather than waiting for the ACK/NACK messages to arrive at the source end. Moreover, this kind of a video transcoder operation can also produce the necessary robust output to counteract the detrimental effects of the video frame drops resulting from network congestions, as introduced in Chapter 3.
Chapter 5  Error-Resilient Video Transcoding

5.4 Computer Simulations and Analysis of the Results

In this section, the computer simulation results are presented to demonstrate the enhancement effects of the error resilience schemes, which were discussed earlier, on the transcoded video quality. Thus, both the objective and subjective results are depicted in such a way that the positive effects of the error-resilient video transcoding over the non-resilient applications are highlighted. The corresponding results obtained from the simulations carried out are discussed in this section for both the AIR and FCS schemes over the transmission channels with burst and random error characteristics. For simulation purposes, the former was modelled with the GPRS channels in which the error characteristics showed mobile fading channel effects with a bursty nature and the latter was designed as random error channels.

5.4.1 Transcoding with AIR

5.4.1.1 Experiments and Results for Random Error Effects

Random error simulations were performed using the 150-frame “Suzie” and the 200-frame “Mother & Daughter” video test sequences. Both sequences were originally encoded with a frame rate of 25 fr/s in I-P-P-P-P-... format. The original bit rate of the “Suzie” sequence prior to the transcoding operation was 94.575 kbit/s on average, giving an average PSNR level of 38.171 dB whilst the bit rate of the “Mother & Daughter” sequence was 70.553 kbit/s on average, giving an average PSNR level of 36.047 dB. The test sequences were encoded, transcoded and decoded in compliance with the MPEG-4 standard with the use of the unrestricted MVs and the advanced prediction modes. “Suzie” was transcoded from 94.575 kbit/s down to 37.838 kbit/s with an overall rate reduction of 60% and ±4-pixel motion vector refinement. Similarly, “Mother & Daughter” was transcoded from 70.553 kbit/s down to 25.818 kbit/s with an overall rate reduction of 63.5% and ±2-pixel motion vector refinement. The frame resolution was chosen as QCIF with 176x144 pixels. The results comprise average PSNR values versus the BERs of a simulated wireless channel. Random bit errors were applied to the VLC indices of the texture and motion data of the transcoded streams rather than directly to the bitstream in order to maintain the synchronisation of the end-decoder with the received video streams. Thus, the video frame headers were assumed to be error-free. Each of the presented test results was obtained from averaging the experimental results of the simulations run with 10 randomly selected seeds. All the simulations were initiated with a pre-determined number of I-MBs for the AIR application, which was set to be a maximum of 3 MBs per frame. However, it should be noted that the number of intra refresh MBs varies with the motion activity and the output transcoded transmission rate variations in an adaptive way, details of which were discussed in the preceding section.
Figure 5.7 Objective results of the 150-frame “Suzie” sequence at near 39 kbit/s on average, a- PSNR versus BER and b- PSNR versus frame number for a BER = 1e-03. (The results shown are the averages of 10 simulations run with different seeds)

Figure 5.8 Objective results of the 200-frame “Mother & Daughter” sequence at near 27 kbit/s on average, a- PSNR versus BER and b- PSNR versus frame number for a BER = 1e-03. (The results shown are the averages of 10 simulations run with different seeds)

Figure 5.7.a demonstrates the simulation results for the “Suzie” sequence whilst Figure 5.8.a presents the results for the “Mother & Daughter” sequence. Both of the figures give the PSNR variations versus the BERs of the video test sequences under the same channel conditions. Both figures illustrate three different variations that represent three different conditions of the same video sequences: error-free, non-resilient error-prone and error-resilient. Moreover, Table 5.2 shows the detailed average PSNR and bit rate values for the different BER conditions for both test sequences, which are presented in the previous two figures.
Chapter 5  
Error-Resilient Video Transcoding

<table>
<thead>
<tr>
<th>Scheme</th>
<th>BER=1e-04</th>
<th>BER=5e-04</th>
<th>BER=1e-03</th>
<th>BER=5e-03</th>
<th>BER=1e-02</th>
</tr>
</thead>
<tbody>
<tr>
<td>150-frame “Suzie”</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>error-free</td>
<td>34.566 dB; 37.838 kbit/s</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>error-prone</td>
<td>33.951; 37.845</td>
<td>31.847; 37.865</td>
<td>30.243; 37.888</td>
<td>25.032; 38.085</td>
<td>22.755; 38.330</td>
</tr>
<tr>
<td>200-frame “Mother &amp; Daughter”</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>error-free</td>
<td>32.683 dB; 25.818 kbit/s</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>error-resilient</td>
<td>32.145; 27.000</td>
<td>30.010; 27.010</td>
<td>30.553; 27.023</td>
<td>26.997; 27.133</td>
<td>24.557; 27.288</td>
</tr>
</tbody>
</table>

Table 5.2  
Average PSNR and bit rate values against different BERs (the first value in each cell before the semi-colon represents the PSNR in dB and the second value shows the bit rate in kbit/s)

Furthermore, Figure 5.7.b and 5.8.b present the objective results for the PSNR variations against the increasing frame numbers for the “Suzie” and “Mother & Daughter” sequences at a chosen BER of 1e-03, respectively. Finally, Figure 5.9 illustrates the subjective results for “Suzie” over the BER = 1e-03 channel for a clear comparison of the non-resilient and the resilient qualities perceptible by the human visual system.

Figure 5.9  
Subjective results of the 150th frames of “Suzie” for a particular seed at BER = 1e-03, a- error-free; b- non-resilient error-prone and c- error-resilient sequences

5.4.1.2 Analysis of the Results

As observed from Figures 5.7.a and 5.8.a, the PSNR characteristics of the different test sequences present similar behaviours; the non-resilient error-prone ones significantly dropping as the BERs increase whilst the error-resilient ones performing much better compared to the non-resilient streams. As expected, the error-free streams remain unaffected by the changing BERs.

From these figures, it can particularly be noticed that the difference between the error-prone results and the error-resilient ones is roughly around 2 dB for the “Suzie” sequence towards the second half of the BER levels. On the other hand, this particular difference is around 4 dB on average for the “Mother & Daughter” sequence. This is due to the fact that the former test sequence was only 150 frames long whereas the latter one was a 200-frame video clip. Since block-based video coding highly depends on predictive encoding and decoding schemes, the
longer the video sequence the more the error propagation takes place. Thus, towards the end of a longer sequence, error accumulation tends to considerably perturb the video quality and hence, the QoS more significantly.

One other possible reason for this discrepancy in results can be the higher motion activity of the “Suzie” sequence compared to its counterpart. Since the number of I-MBs was limited to a maximum of fixed 3 MBs, the quality improvement in “Suzie” with higher motion activity was not fully achieved compared to the more improved results of the “Mother & Daughter” sequence. As it is clearly deduced from the discussion of the theoretical AIR methodology in Chapter 2, the more the I-MBs the better the quality in error-prone conditions at the expense of a higher output bit rate. However, it is important to note that the aim of error-resilient video transcoding is not only to produce a resilient output stream, but also a video stream whose transmission rate characteristics match the requirements and/or constraints of the host standards. Therefore, the goal has been fulfilled with the error-resilient transcoder output at a near target average bit rate.

Furthermore, as noticed from Table 5.2, as the BER increases, in other terms as the channel conditions worsen, the average bit rates tend to increase with very small amounts. This is due to the fact that for simulation purposes, the error conditions were indeed applied to the VLC indices rather than directly to the bitstream, as also explained earlier. Therefore, as the error rate increases, the number of corrupted indices increases resulting in longer codewords that eventually increase the overall bit rate. However, as also noted from the table, the particular increase in the bit rate due to this fact is relatively insignificant amounting to a maximum of 0.485 kbit/s.

Table 5.2 also clearly presents the efficiency of the proposed and tested error-resilient video transcoding scheme. The transcoded quality improvement obtained was as 2 dB on average for the “Suzie” and 4 dB on average for the “Mother & Daughter” sequences, as also stated formerly. However, it is important to note that these quality improvements were achieved with the minimum amount of bit rate increases at the output which were 1.5 kbit/s for the “Suzie” and 1.7 kbit/s for the “Mother & Daughter” sequences. From these improved PSNRs and insignificantly changed bit rates, it can be deduced that the error-resilient transcoder outperforms the error-prone schemes whilst giving near demanded transmission bit rates.

The differences in average quality, in terms of PSNRs of the transcoded video streams between the error-prone and the error-resilient cases, were 2 dB for “Suzie” and 2.3 dB for “Mother & Daughter” at the chosen BER = 1e-03, as also observed from Figures 5.7.b and 5.8.b. Meanwhile, the increase in the bit rates was reported to be as little as 1 kbit/s in both cases. The detrimental impact of the channel errors and the quality improvement by the resilient transcoding can clearly be seen in the 150th frames of the “Suzie” illustrated in Figure 5.9. In this figure, the subjective results demonstrate that the particular instantaneous PSNRs are 34.354 dB for the error-free,
32.471 dB for the error-resilient and 24.705 dB for the non-resilient error-prone cases. On the other hand, the obtained bit rates were -38 kbit/s, -38 kbit/s and -39 kbit/s for the error-free, error-prone and error-resilient operations, respectively.

5.4.1.3 Experiments and Results for Burst Error Effects

The burst error effects were simulated over the GPRS channel simulator which was genuinely designed and implemented within the Centre for Communication Systems Research (CCSR). In terms of error effects, the characterisation of a GPRS channel is modelled as a bursty error-prone transmission environment where fairly big chunks of the transmitted data becomes highly susceptible to the detrimental error impacts [GSM0503, GSM0505]. This kind of errors corrupt the conveyed information more significantly than the random error effects as far as QoS is concerned. This impact particularly destroys the video communication data since even a single bit error, in the form of a bit loss or an inversion, leads to a serious synchronisation problem or a rapidly increasing and spreading error propagation within the transmitted video sequence. Thus, the error propagation has to be stopped and the synchronisation has to be resumed during the transmission of the video information. Therefore, this sub-section is dedicated to the investigation of the GPRS channel effects on video transcoding.

In this sub-section, two different 200-frame video sequences were tested over a GPRS channel model. The two test sequences were deliberately chosen to comprise two different motion activity natures: “Mother & Daughter” and “Foreman” with moderate and high activity scenes, respectively. Both test sequences were encoded, transcoded and decoded in compliance with the MPEG-4 standard with the use of the unrestricted MVs and the advanced prediction modes. The frame rates, frame sizes and the operation modes were set to 25 fr/s, QCIF (176×144 pixels) and I-P-P-P-... format for both video clips, respectively. The original bit rate of the 200-frame “Mother & Daughter” sequence prior to the transcoding operation was 70.553 kbit/s on average, giving an average PSNR level of 36.047 dB. This sequence was later transcoded down to an average rate of 25.818 kbit/s with a PSNR level of 32.683 dB. Similarly, the 200-frame “Foreman” sequence was transcoded from an average rate of 87.403 kbit/s with a PSNR level of 33.582 dB down to 46.835 kbit/s on average with a PSNR level of 30.029 dB. Thus, the rate reductions applied on “Mother & Daughter” and “Foreman” were 63.5% and 46.5%, respectively. Moreover, the MV refinement window sizes were set to ±2 pixels and ±5 pixels for “Mother & Daughter” and “Foreman”, respectively.

The bit rate reductions were essential to enable the video streams to transport over the GPRS channels in such a typical video communication scenario, as depicted in Figure 5.4. As the last column of Table 5.1 clearly indicates, the amount of user data for the transport over GPRS is
strictly limited depending on the selected channel protection scheme. However, the timeslotting feature of GPRS can overcome this kind of limitation to some extent. Nevertheless, despite the multi-slotting feature, GPRS rates are still far too low for video communications if any frame droppings are not employed. Therefore, a successful error-resilient video transcoding for transmission rate reduction is necessary prior to the GPRS network transport, as also illustrated in Figure 5.4. Multiple slots can be used to further increase the user bit rate as multiples of the base transmission rate, as depicted in Table 5.3. In this table, the first column has been illustrated with a shaded pattern as to describe that following slots are multiples of the data rates given in this first column. Although this particular table seems to indicate different user data rates for different channel protection schemes from the figures given in Table 5.1, there is indeed not any kind of mismatches between these particular two tables. This is only due to the fact that actual raw application level user rates for the user applications are given in Table 5.3 whereas Table 5.1 also comprises the added overheads. Naturally, Table 5.1 user rates are slightly higher than those of Table 5.3. However, it has to be denoted that the raw data rates presented in Table 5.3 were obtained from a series of video transmissions over GPRS with various test sequences; they do not constitute a part of the GPRS standard. During the GPRS simulations presented in this subsection, this particular table, namely Table 5.3, guided the selection of the transcoded raw user video rates as at the application layer. Thus, this kind of a lower transcoding rate selection enabled the simulation results to become more realistic as more overheads would be added to the produced raw transcoding rates through the protocol stack. Furthermore, channel protection schemes in terms of various convolutional code rates would also be added to the overall data rate which also increased the transmission rate on the whole.

<table>
<thead>
<tr>
<th>Timeslots</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS1</td>
<td>6800</td>
<td>13600</td>
<td>20400</td>
<td>27200</td>
<td>34000</td>
<td>40800</td>
<td>47600</td>
<td>54400</td>
</tr>
<tr>
<td>CS2</td>
<td>10500</td>
<td>21000</td>
<td>31500</td>
<td>42000</td>
<td>52500</td>
<td>63000</td>
<td>73500</td>
<td>84000</td>
</tr>
<tr>
<td>CS3</td>
<td>12200</td>
<td>24400</td>
<td>36600</td>
<td>48800</td>
<td>61000</td>
<td>73200</td>
<td>85400</td>
<td>97600</td>
</tr>
<tr>
<td>CS4</td>
<td>17200</td>
<td>34400</td>
<td>51600</td>
<td>68800</td>
<td>86000</td>
<td>103200</td>
<td>120400</td>
<td>137600</td>
</tr>
</tbody>
</table>

Table 5.3 Timeslotting capability and the raw user data rates employed for the experimentation of the different GPRS channel protection schemes

As similar to the simulations presented in the former sub-sections for the transmission channels with random error characteristics, AIR is also provided here as the major error resilience tool on the transcoded video streams. Similarly, all the simulations were initiated with a pre-determined number of I-MBs, which was set to be a maximum of 3 MBs per frame. However, it should also be indicated that the number of intra refresh MBs varies with the motion activity and the output
transcoded transmission rate variations in an adaptive way, details of which were discussed in the preceding section.

Despite a number of similarities in the experimental set up of the GPRS simulations to the random error simulations presented earlier, there are also significant discrepancies, such as the operation of the very end-decoder. In the previous sub-sections, random errors were only introduced to the video data rather than the video frame headers to maintain the synchronisation. On the other hand, since the GPRS channel model corrupts the entire bitstream which is fed through, bursty errors can possibly cause a number of synchronisation losses while decoding. Thus, a few but minor modifications were inevitable in the operation of the end-decoder. These modifications enabled the end-decoder to maintain the synchronisation. These modifications comprised the simple error detection capabilities, as described in Chapter 2, and skipping of the following video stream until a new start code or a synchronisation word was detected. Following these two significant operations, the decoder resumed decoding from that particular point onwards rather than stopping and giving up the decode and display processes, as usually happens in the traditional operations. However, it is also worthwhile to mention that instead of the skipped and discarded video data, the decoder applied some simple form of video data concealment. The concealment was accomplished through setting the discarded MBs of an erroneous video frame to become uncoded MBs. Following this process, the motion compensation gave smoother results as the texture data of the spatially corresponding MBs in the previous video frame were re-used for the discarded MB data [Wang]. This achievement was necessary at the end-decoder for a successful decode operation as the source coding MPEG-4 simulation software was operated without the use of any error resilience options.

Simulation results, each of which was obtained from averaging the experimental results of 10 different random seeds, are depicted in Figures 5.10-13 for both objective and subjective comparisons. The first set of objective and subjective results was obtained for the "Mother & Daughter" sequence and the second set was obtained for the "Foreman" sequence. All the results presented in this sub-section comprise the simulations using three different channel protection schemes as the fourth scheme (CS4) is not practically feasible for video applications [Cella, Fabri]. Therefore, the results demonstrate the non-resilient error-prone and error-resilient transcoding applications along with the results of the error-free sequences for comparative referencing purposes.
Figure 5.10 Objective results of the 200-frame “Mother & Daughter” sequence at near 27 kbit/s on average for CS1, CS2 and CS3, a- PSNR versus C/I and b- PSNR versus frame number for a C/I = 12 dB. (The results shown are the averages of 10 simulations run with different seeds)

![Graph a](image1)

![Graph b](image2)

Table 5.4 Average PSNR and bit rate values against different C/I ratios

<table>
<thead>
<tr>
<th>Scheme</th>
<th>C/I=7 dB</th>
<th>C/I=9 dB</th>
<th>C/I=12 dB</th>
<th>C/I=15 dB</th>
<th>C/I=18 dB</th>
<th>Rate [kbit/s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>error-free</td>
<td>32.683 dB</td>
<td>25.818</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>error-resilient</td>
<td>21.8111 dB</td>
<td>27.2557 dB</td>
<td>31.5382 dB</td>
<td>32.683 dB</td>
<td>N/A</td>
<td>27.000</td>
</tr>
<tr>
<td>CS2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>error-free</td>
<td>32.683 dB</td>
<td>25.818</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CS3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>error-free</td>
<td>32.683 dB</td>
<td>25.818</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 5.5 Experimentally obtained BERs against C/I for “Mother & Daughter” for different channel protection schemes over GPRS

<table>
<thead>
<tr>
<th>C/I [dB]</th>
<th>Scheme</th>
<th>BER</th>
<th>Scheme</th>
<th>BER</th>
<th>Scheme</th>
<th>BER</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>C</td>
<td>1.083e-02</td>
<td>C</td>
<td>5.006e-02</td>
<td>C</td>
<td>9.686e-02</td>
</tr>
<tr>
<td>9</td>
<td>S</td>
<td>2.704e-03</td>
<td>S</td>
<td>2.047e-02</td>
<td>S</td>
<td>5.144e-02</td>
</tr>
<tr>
<td>12</td>
<td>S</td>
<td>2.953e-04</td>
<td>S</td>
<td>3.323e-03</td>
<td>S</td>
<td>1.360e-02</td>
</tr>
<tr>
<td>15</td>
<td>1</td>
<td>0.0000000</td>
<td>2</td>
<td>4.440e-04</td>
<td>3</td>
<td>1.968e-03</td>
</tr>
<tr>
<td>18</td>
<td>N/A</td>
<td>2.505e-04</td>
<td>3</td>
<td>1.446e-04</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 5.5 Experimentally obtained BERs against C/I for “Mother & Daughter” for different channel protection schemes over GPRS
Figure 5.11 Subjective results of the 200th frames of "Mother & Daughter" for a particular seed at C/I = 12 dB, a- error-free direct enc/dec at 70.553 kbit/s; b- CS1, non-resilient error-prone transoded down to 25.818 kbit/s; c- CS2, non-resilient error-prone transcoded down to 25.818 kbit/s; d- CS3, non-resilient error-prone transcoded down to 25.818 kbit/s; e- error-free transcoded down to 25.818 kbit/s; f- CS1, error-resilient transcoded down to 27.000 kbit/s; g- CS2, error-resilient transcoded down to 27.000 kbit/s; h- CS3, error-resilient transcoded down to 27.000 kbit/s sequences

Figure 5.10.a demonstrates the PSNR results of "Mother & Daughter" over a varying carrier-to-interference (C/I) ratio whilst Figure 5.10.b presents the PSNR variations versus video frame numbers for a particular operation point of C/I = 12 dB for CS1, CS2 and CS3 coding schemes. The necessary number of timeslots for these three channel protection schemes are clearly depicted within the presented results. The timeslots were adequately chosen depending on the produced video rates during the transcoding processes in reference to Table 5.3. Tables 5.4-5 present more detailed results for PSNR versus C/I and BER versus C/I, respectively. Moreover, Figure 5.11 illustrates the subjective results of the 200th frames of "Mother & Daughter" at C/I = 12 dB for the different GPRS channel protection schemes, namely CS1, CS2 and CS3.
Figure 5.12 Objective results of the 200-frame “Foreman” sequence at near 47 kbit/s on average for CS1, CS2 and CS3. a- PSNR versus C/I and b- PSNR versus frame number for a C/I = 12 dB.

(The results shown are the averages of 10 simulations run with different seeds)

<table>
<thead>
<tr>
<th>Scheme</th>
<th>C/I=7 dB</th>
<th>C/I=9 dB</th>
<th>C/I=12 dB</th>
<th>C/I=15 dB</th>
<th>C/I=18 dB</th>
<th>Rate [kbit/s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>error-free</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>46.835</td>
</tr>
<tr>
<td>error-prone</td>
<td>17.7749 dB</td>
<td>20.2871 dB</td>
<td>26.3716 dB</td>
<td>30.029 dB</td>
<td>N/A</td>
<td>46.835</td>
</tr>
<tr>
<td>error-resilient</td>
<td>18.5075 dB</td>
<td>22.4286 dB</td>
<td>28.6215 dB</td>
<td>30.029 dB</td>
<td>N/A</td>
<td>46.986</td>
</tr>
<tr>
<td>CS2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>46.835</td>
</tr>
<tr>
<td>error-prone</td>
<td>15.2017 dB</td>
<td>17.0912 dB</td>
<td>22.1539 dB</td>
<td>27.2675 dB</td>
<td>29.0246 dB</td>
<td>46.986</td>
</tr>
<tr>
<td>CS3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>46.835</td>
</tr>
</tbody>
</table>

Table 5.6 Average PSNR and bit rate values against different C/I ratios

<table>
<thead>
<tr>
<th>C/I [dB]</th>
<th>Scheme</th>
<th>BER</th>
<th>Scheme</th>
<th>BER</th>
<th>Scheme</th>
<th>BER</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>C</td>
<td>1.218e-02</td>
<td>C</td>
<td>5.420e-02</td>
<td>C</td>
<td>1.072e-01</td>
</tr>
<tr>
<td>9</td>
<td>S</td>
<td>2.923e-03</td>
<td>S</td>
<td>2.147e-02</td>
<td>S</td>
<td>5.575e-02</td>
</tr>
<tr>
<td>12</td>
<td>1</td>
<td>1.999e-04</td>
<td>2</td>
<td>3.496e-03</td>
<td>3</td>
<td>1.443e-02</td>
</tr>
<tr>
<td>15</td>
<td>0.0000000</td>
<td>3.519e-04</td>
<td>3</td>
<td>2.054e-03</td>
<td>3</td>
<td>1.791e-04</td>
</tr>
<tr>
<td>18</td>
<td>N/A</td>
<td>9.424e-05</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 5.7 Experimentally obtained BERs against C/I for “Foreman” for different channel protection schemes over GPRS
Chapter 5

Figure 5.13 Subjective results of the 200th frames of “Foreman” for a particular seed at C/I = 12 dB, a- error-free direct enc/dec at 87.403 kbit/s; b- CS1, non-resilient error-prone transcoded down to 46.835 kbit/s; c- CS2, non-resilient error-prone transcoded down to 46.835 kbit/s; d- CS3, non-resilient error-prone transcoded down to 46.835 kbit/s; e- error-free transcoded down to 46.835 kbit/s; f- CS1, error-resilient transcoded down to 46.986 kbit/s; g- CS2, error-resilient transcoded down to 46.986 kbit/s; h- CS3, error-resilient transcoded down to 46.986 kbit/s.

Finally, Figure 5.12.a demonstrates the PSNR results of “Foreman” over a varying C/I ratio whilst Figure 5.12.b presents the PSNR variations versus sequence frame numbers for a particular operation point of C/I = 12 dB for CS1, CS2 and CS3 coding schemes. The necessary number of timeslots for these three channel protection schemes are also depicted within the presented results. Similarly, the timeslots were adequately chosen depending on the produced video rates during the transcoding processes in reference to Table 5.3. Tables 5.6-7 also present more detailed results for PSNR versus C/I and BER versus C/I, respectively. Furthermore, Figure 5.13 illustrates the subjective results of the 200th frames of “Foreman” at C/I = 12 dB for the various GPRS channel protection schemes, namely CS1, CS2 and CS3.

5.4.1.4 Analysis of the Results

As observed from the two sets of experimental results, the error-resilient video transcoding performance over the GPRS channel model presented improved video qualities compared to the non-resilient scheme. This performance improvement is particularly notable from Figures 5.10.a and 5.12.a. These particular figures demonstrate the various average quality levels achieved whilst the experiments were conducted for the different CS schemes for “Mother & Daughter” and
“Foreman”, respectively. Furthermore, Tables 5.4 and 5.6 also contribute to the performance comparisons of the error-resilient and non-resilient operations of both test sequences. To allow for an easier and clearer understanding of the simulation results, Table 5.8 is also depicted to present the detailed quality improvement obtained during the tests.

<table>
<thead>
<tr>
<th>Scheme</th>
<th>C/I = 7 dB</th>
<th>C/I = 9 dB</th>
<th>C/I = 12 dB</th>
<th>C/I = 15 dB</th>
<th>C/I = 18 dB</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>200-frame “Mother &amp; Daughter” at near 27 kbit/s on average</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CS1</td>
<td>-2 dB</td>
<td>-3 dB</td>
<td>-1 dB</td>
<td>0 dB</td>
<td>N/A</td>
</tr>
<tr>
<td>CS2</td>
<td>-1 dB</td>
<td>-1 dB</td>
<td>-3 dB</td>
<td>-2 dB</td>
<td>-0.2 dB</td>
</tr>
<tr>
<td>CS3</td>
<td>-0.6 dB</td>
<td>-1 dB</td>
<td>-2 dB</td>
<td>-2.6 dB</td>
<td>-2 dB</td>
</tr>
<tr>
<td></td>
<td>200-frame “Foreman” at near 47 kbit/s on average</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CS1</td>
<td>-1 dB</td>
<td>-2 dB</td>
<td>-2 dB</td>
<td>0 dB</td>
<td>N/A</td>
</tr>
<tr>
<td>CS2</td>
<td>-0.4 dB</td>
<td>-0.3 dB</td>
<td>-2 dB</td>
<td>-2 dB</td>
<td>-0.5 dB</td>
</tr>
<tr>
<td>CS3</td>
<td>-0.1 dB</td>
<td>-0.2 dB</td>
<td>-1 dB</td>
<td>-1.5 dB</td>
<td>-1.5 dB</td>
</tr>
</tbody>
</table>

Table 5.8 The video quality improvements by the error-resilient transcoding over the non-resilient scheme

Table 5.8 demonstrates that the error-resilient “Mother & Daughter” sequence performed slightly better than the error-resilient “Foreman” sequence for the all three CS conditions. This outcome implies that the high motion activity of “Foreman” might have imposed a limitation over the performance improvement within the significantly perturbed transmission conditions. Thus, the bursty error effects on the transmitted video presented similar behaviour as the random error effects. However, the degree of the video quality degradation is much more distinguished here as the destruction effects of the burst errors are fairly critical to the error-sensitive video data. Particularly, the objective video qualities, which have been demonstrated in Figures 5.10.a and 5.12.a, are unacceptable for “Mother & Daughter” with CS3 and for “Foreman” with CS2 and CS3 at C/I = 7 dB (PSNR: below 15 dB). At this particularly very low C/I ratio, it has also been noticed that the sole 3-MB AIR resilience method did not perform satisfactorily for either of the test video clips. In addition, the error-resilient “Foreman” sequence also presented similar low quality results for CS2 and CS3 at C/I = 9 dB. However, this is not the case for “Mother & Daughter” at C/I = 9 dB as the error sensitivity of the high motion scene plays an important role over the QoS in error-prone conditions.

Thus, it is recommended to employ a combination of suitable error resilience tools rather than the sole AIR scheme at these particularly very low C/I ratios over GPRS. On the other hand, the AIR method presented quite satisfactory performance improvements at various other C/I ratios and with different CS schemes, as seen in Table 5.8 and Figures 5.10.a and 5.12.a. Naturally, for low BERs, or in other terms high C/I ratios (i.e. C/I = 18 dB), quality improvement features of the error resilience methods are limited. The experimental BERs versus different C/I ratios can be
seen in Tables 5.5 and 5.7 for "Mother & Daughter" and "Foreman", respectively. It is clear from these tables that as C/I decreases, the BER increases.

The effectiveness of the error-resilient video transcoding can also be observed in Figures 5.10.b and 5.12.b. These particular figures demonstrate that the performance of the resilient schemes are superior over the non-resilient schemes for a chosen C/I = 12 dB for the "Mother & Daughter" and "Foreman" sequences, respectively. This outcome does not contradict the other previously presented and discussed results, but supports them. It is worthwhile pointing out here that the video quality degradation between the CS1 and CS2 results is quite notable for both of the sequences which sums up to 6 dB on average. This shows that the destructive effects of the increasing BER with the variation of the different CS conditions.

Finally, Figures 5.11 and 5.13 illustrate the GPRS channel effects on the non-resilient and error-resilient transcoded video for both of the test sequences. These figures depict the 200th frames of "Mother & Daughter" and "Foreman" video clips for three different CSs at C/I = 12 dB, as the complementary subjective results to the objective ones presented in Figures 5.10.b and 5.12.b. The figures clearly depict that the error-resilient transcoding results show perceptible improvement in the video service quality performance. This significant improvement was achieved at near target bit rates despite the fact that the AIR method imposed an output transmission rate increase. However, with the exploitation of the useful rate reduction features of the video transcoder, bit rate management was no longer means of a trade-off issue. The error resilience was thus introduced to the compressed video streams at an intermediate level within the entire communication network at the expense of merely 1 kbit/s and 0.1 kbit/s growths for "Mother & Daughter" and "Foreman", respectively. The obtained near target bit rates were 27 kbit/s on average for "Mother & Daughter" and 47 kbit/s on average for "Foreman". These particular rates allowed the former to be transmitted over 4 CS1, 3 CS2 or CS3 timeslots and the latter to be conveyed over 7 CS1, 5 CS2 or 4 CS3 timeslots via the GPRS access network.

5.4.2 Transcoding with FCS

5.4.2.1 Experiments and Results with Heterogeneous Transcoding

The heterogeneous video transcoding has been described and shown to be necessary to link two video coding standards, such as H.263 and MPEG-4, in Chapter 4. This involves a much simpler approach than firstly decoding the incoming bitstream at the video gateway and then re-encoding it with the syntax of the new standard. Otherwise, it is a very time and power consuming operation with the added complexity and the reduced picture quality [Dogan4].
The heterogeneous transcoding can easily overcome these problems by the translation and mapping of the syntax of the relevant video standards, as discussed in the previous chapter. However, it is also important that the entire video communication system is made robust to possible channel errors as channel errors severely degrade the two-way video communication quality. For this reason, this sub-section considers not only the straightforward heterogeneous transcoding algorithm, but also some intelligence added to the heterogeneous video transcoder, in terms of error resilience.

![Figure 5.14 Error-resilient heterogeneous video gateway application scenarios](image)

The associated problem has been based on the operation of the MPEG-4 standard mainly over the satellite and mobile-wireless channels for multimedia applications and the H.263 standard on the ISDNs/PSTNs, as illustrated in Figure 5.14. Therefore, the BER is accepted to be much less for the H.263 operation in circuit-switched networks than for the MPEG-4 applications in wireless highly error-prone environments. Thus, the error resilience algorithm is primarily applied onto the MPEG-4 video as it is accepted to encounter higher BER and more notable varying delay effects which degrade the two-way video communications quality significantly.

As previously described in Chapter 2, FCS is a technique that relies on a back channel signal which informs the encoder of the lost or the properly delivered video frames. Thus, this particular feedback signal helps the encoder adapt its encoding scheme according to the varying channel conditions and/or constraints. However, in this sub-section, this feedback signal is sent from the video gateway, which comprises the heterogeneous video transcoder, back to the source encoder unlike in the conventional feedback controlling scenarios where the control signal is sent all the way back from the decoder at the receiving end. This feature gives some intelligence to the transcoder as it can follow the temporal reference, decide which frames are lost or received over the error-prone channels and signal them back to the encoder. Thus, the feedback signalling path becomes shortened as the feedback signal does not arrive at the encoder from the decoder, which is located at the very end of the entire communication system, but from a much earlier point: the
Clearly, this feature of the transcoder saves a very good amount of time for the source to respond to the varying error or network conditions. Not only does the feedback signalling from the video transcoder resolve the problems related to the error-prone conditions, but also the network congestion situations are catered for. Similarly, the transcoder can also send a backward signal to report any congestion occurrences over either the ISDN/PSTN or the MPEG-4 network and inform the encoder of the network conditions.

As the video stream is conveyed towards the H.263 side, the error ratios and the delay effects decrease to insignificant levels. On the way back, which is H.263 towards MPEG-4, none of the resilience techniques are applied to the video stream for this particular research work presented in this section. However, the resilience algorithms can be elaborated for the reverse path as well. In such a case, the feedback signal will return from the MPEG-4 decoder rather than the video transcoder itself which eliminates the intelligence added onto the transcoder. In this particular case, neither the transcoder nor the video gateway, but the MPEG-4 decoder is the only informer of the channel conditions. Therefore, it should be noted that the feedback signal will further be delayed causing some additional significant corruptive effects on the two-way video communications quality.

In this sub-section, the experiments were performed with two video test sequences: "Suzie" with 150 frames and "Claire" with 200 frames. Both video sequences were encoded in I-P-P-P-P-... format at 25 fr/s with the QCIF frame size comprising 176x144 pixels.

**Figure 5.15** Objective test results of "Suzie" (the lost frame numbers: 9, 17, 52, 77, 100, 133, 142), a- PSNR and b- bit rate variations
Chapter 5  
Error-Resilient Video Transcoding

MPEG-4 $\rightarrow$ H.263 transcoded

![Graph](image)

$a$

Figure 5.16 Objective test results of “Claire” (the lost frame numbers: 13, 34, 69, 111, also 112 for the 2-consecutive frame loss case, 145, 163, 178), $a$- PSNR and $b$- bit rate variations

![Video Frames](image)

$a$

Figure 5.17 Subjective results of the 56th frames of “Suzie”, $a$- error-free; $b$- non-resilient error-prone (5% frame loss) and $c$- 2-frame delay error-resilient (5% frame loss) sequences

![Video Frames](image)

$a$

Figure 5.18 Subjective results of the 116th frames of “Claire”, $a$- error-free; $b$- non-resilient error-prone (3.5% frame loss); $c$- non-resilient error-prone with 2-consecutive frame loss and $d$- 2-frame delay error-resilient (3.5% frame loss) sequences

140
The objective and subjective results are presented in this sub-section only for the transcoded video streams as the quality degradation of the transcoded streams from the reference (direct encoded/decoded) streams has already been proved to be negligible in Chapter 4.

The results are presented for the error-free, the non-resilient error-prone (with the 3.5% frame loss for “Claire” and the 5% frame loss for “Suzie”) and the error-resilient with 2-frame delay cases. Furthermore, the investigation of the effect of a 2-consecutive frame loss result is also presented.

Figures 5.15 and 5.16 depict the objective results whilst Figures 5.17 and 5.18 illustrate the subjective results for the “Suzie” and “Claire” sequences, respectively. The numbers of the randomly chosen and deliberately lost frames, in order to simulate the error-prone channel with frame loss characteristics, are also notified within the associated figure captions. Moreover, Table 5.9 demonstrates the average PSNRs and the bit rates of the simulated various schemes in more detail.

<table>
<thead>
<tr>
<th>Type of the Scheme</th>
<th>Av. Bit Rate [kb/s]</th>
<th>Av. PSNR [dB]</th>
</tr>
</thead>
<tbody>
<tr>
<td>150-frame “Suzie”</td>
<td></td>
<td></td>
</tr>
<tr>
<td>error-free</td>
<td>56.948</td>
<td>34.565</td>
</tr>
<tr>
<td>non-resilient error-prone</td>
<td>56.948</td>
<td>27.869</td>
</tr>
<tr>
<td>error-resilient with 2-frm delay</td>
<td>60.554</td>
<td>34.100</td>
</tr>
<tr>
<td>200-frame “Claire”</td>
<td></td>
<td></td>
</tr>
<tr>
<td>error-free</td>
<td>24.895</td>
<td>36.333</td>
</tr>
<tr>
<td>non-resilient error-prone</td>
<td>24.895</td>
<td>35.635</td>
</tr>
<tr>
<td>error-prone, 2-consecutive frm loss</td>
<td>24.895</td>
<td>35.510</td>
</tr>
<tr>
<td>error-resilient with 2-frm delay</td>
<td>29.333</td>
<td>36.223</td>
</tr>
</tbody>
</table>

Table 5.9 Average bit rate and PSNR values for the two test sequences

5.4.2.2 Analysis of the Results

As observed from Figure 5.15, the 5% frame loss condition has a very detrimental impact on the picture quality as the very high motion frame (frame number 52) of “Suzie” was lost during the transmission. This deteriorating effect could not be compensated for the error-prone condition without the use of error-resilience scheme, leading to a continuous degraded quality throughout the rest of the entire transmission. On the contrary, error-resilient encoding showed to be able to compensate for the quality degradation back to the error-free PSNR levels, as demonstrated in Figure 5.15.

Similar results were obtained for the “Claire” sequence, as presented in Figure 5.16. In this figure, an additional PSNR variation can be noticed, which is the 2-consecutive frame loss case. This situation resulted in a more degraded picture quality, in terms of PSNR levels, than the usual random frame loss cases. This simply happens due to the lack of correlation of the predicted frames which were decoded at the receiver. However, as also seen in the figure, the resilient sequence was not affected as the loss of the first frame of the consecutive two frames was already
compensated with the help of the feedback signal. In both figures, the bit rate variations gave similar peaks as in FCS results of Chapter 2 which indicate that the relevant video frames were encoded in a resilient mode.

Furthermore, the subjective results, which are illustrated in Figure 5.17 and 5.18, highly support the analysis of the objective results. Particularly examining the "Suzie" pictures, the quality deterioration and the improvement with the use of the resilience algorithm can clearly be noticed. This is due to the reason that the 56th frame presents the detrimental effect of the loss of the 52nd frame which indeed had a very high motion activity. This particular effect can also be perceived from the corresponding PSNR graph of the "Suzie" sequence in Figure 5.15. The 116th frames of the "Claire" sequences also highlight the deteriorating effects of the loss of one or two consecutive frames and the improvement in the picture quality that can be perceived by the use of the error resilience scheme. The reason for displaying, particularly the 116th frame is that this frame clearly shows the impacts of the 2 consecutive frame losses on the picture quality and the improvement that was achieved by the resilience algorithm.

Finally, the detailed average PSNR and the bit rate results of the both test sequences show that the quality difference between the resilient and the non-resilient error-prone cases is around 6 dB for "Suzie", as presented in Table 5.9. This amount of quality improvement is shown to be achieved at the expense of only a 3.6 kbit/s bit rate increase. On the other hand, the quality improvement for "Claire" is reported to be as little as 0.7 dB with a rate increase of 4.4 kbit/s in the table. This is due to the fact that "Suzie" contains much higher motion activity than "Claire" as a test sequence and hence, frame loss effects are observed to be more significant for this particular sequence. Thus, quality improvement features of the error resilience scheme is obtained to be more perceivable in comparison to a test sequence with a moderate motion activity, such as "Claire".

5.4.2.3 Experiments and Results with Homogeneous Transcoding

The FCS experiments were carried out in two major parts. The first part was designed to simulate the effects of frame losses and the FCS resilience operation at various ACK/NACK reception round-trip delay conditions. The different transcoded video performances were tested for the back channel signal reception times of up to 480 msec, which coincide with the duration of 12 transcoded video frames at the frame rate of 25 fr/s. The second part was set up to investigate the effects of significantly long round-trip delays of the ACK/NACK signal over a GPRS mobile-access network. This particular end-to-end round-trip delay was reported to be ~450 msec (11.25 video frames at 25 fr/s) in the phase 1 of the initial GPRS standard [GSM0364]. Thus, the experimental set-up was built in such a way that a loss of a GPRS radio packet is reported back to the video gateway from a receiving end-terminal in 450 msec, as depicted in Figure 5.19. This
round-trip delay refers to the time elapsed whilst waiting for an ACK or NACK to arrive back at the homogeneous video transcoder. Meanwhile, the transcoder keeps on processing the input video frames at the frame rate of 25 fr/s and hence, the gateway carries on transmitting the transcoded video frames in GPRS radio packets. The assumption made here for the GPRS access network experiments is that one video frame fits into one GPRS radio packet prior to the transmission. Therefore, the loss of a GPRS packet is directly related to the loss of a video frame for a simplified simulation model. However, on a few occasions during the tests, two consecutive video frame losses were also experienced which were assumed to fit in one GPRS radio packet.

![GPRS Network Diagram](image)

**Figure 5.19** Round-trip delay of the feedback signal over GPRS

![Graphs](image)

**Figure 5.20** Objective results of “Suzie”, average PSNR variations against  
**a**- feedback signalling round-trip delay and  
**b**- number of transcoded frames
Figure 5.21 Objective results of “Salesman”, average PSNR variations against \( a \)- feedback signalling round-trip delay and \( b \)- number of transcoded frames

Figure 5.22 Objective results of “Foreman”, average PSNR variations against \( a \)- feedback signalling round-trip delay and \( b \)- number of transcoded frames

The objective and subjective results for the first set of frame loss and the remedial FCS experiments are demonstrated in Figures 5.20-23 and Table 5.10. This particular set comprises the simulation results of the 150-frame “Suzie”, 150-frame “Salesman” and 200-frame “Foreman” video test sequences. Objective results include the average PSNR variations against the various round-trip delay conditions for the ACK/NACK reception and against the number of transcoded video frames. The reason why the PSNR results show an average is that each of the simulations represents an average outcome of 10 different simulations run with 10 different random seeds. Table 5.10 presents the detailed quality levels and the changes in bit rates imposed by the added...
resilience. The subjective results illustrate the last frames taken from each of the experimented video test clips with different frame delays for the resilient and non-resilient cases as well as the error-free ones provided for reference.
Figure 5.23 Subjective results of the 150th frames of “Suzie” and “Salesman” and 200th frames of “Foreman”, respectively. a- error-free; b- 2-frame delay; c- 6-frame delay; d- 10-frame delay resilient; e- error-prone; f- 4-frame delay; g- 8-frame delay; h- 12-frame delay resilient sequences.

<table>
<thead>
<tr>
<th>Type of the Scheme</th>
<th>150-frame “Suzie”, MV refinement window size: ±2 pixels</th>
<th>200-frame “Foreman”, MV refinement window size: ±5 pixels</th>
</tr>
</thead>
<tbody>
<tr>
<td>direct enc/dec @ high bit rate</td>
<td>172.685</td>
<td>87.403</td>
</tr>
<tr>
<td>transcoded error-free</td>
<td>46.353</td>
<td>46.835</td>
</tr>
<tr>
<td>transcoded non-resilient error-prone</td>
<td>46.353</td>
<td>46.835</td>
</tr>
<tr>
<td>transcoded 2-frm delay error-resilient</td>
<td>47.207</td>
<td>47.168</td>
</tr>
<tr>
<td>transcoded 4-frm delay error-resilient</td>
<td>47.237</td>
<td>47.036</td>
</tr>
<tr>
<td>transcoded 6-frm delay error-resilient</td>
<td>47.305</td>
<td>47.036</td>
</tr>
<tr>
<td>transcoded 8-frm delay error-resilient</td>
<td>47.521</td>
<td>47.411</td>
</tr>
<tr>
<td>transcoded 10-frm delay error-resilient</td>
<td>47.912</td>
<td>47.117</td>
</tr>
<tr>
<td>transcoded 12-frm delay error-resilient</td>
<td>48.410</td>
<td>47.018</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Av. Bit Rate [kbit/s]</th>
<th>Av. PSNR [dB]</th>
</tr>
</thead>
<tbody>
<tr>
<td>40.323</td>
<td>35.446</td>
</tr>
<tr>
<td>29.311</td>
<td>31.749</td>
</tr>
<tr>
<td>31.761</td>
<td>31.737</td>
</tr>
<tr>
<td>31.643</td>
<td>31.434</td>
</tr>
<tr>
<td>31.715</td>
<td>31.715</td>
</tr>
<tr>
<td>37.383</td>
<td>34.991</td>
</tr>
<tr>
<td>33.445</td>
<td>33.576</td>
</tr>
<tr>
<td>33.541</td>
<td>33.565</td>
</tr>
<tr>
<td>33.589</td>
<td>33.634</td>
</tr>
<tr>
<td>33.690</td>
<td>33.690</td>
</tr>
<tr>
<td>33.582</td>
<td>30.029</td>
</tr>
<tr>
<td>27.732</td>
<td>28.099</td>
</tr>
<tr>
<td>27.923</td>
<td>28.034</td>
</tr>
<tr>
<td>28.053</td>
<td>28.053</td>
</tr>
<tr>
<td>27.989</td>
<td>27.989</td>
</tr>
<tr>
<td>28.021</td>
<td>28.021</td>
</tr>
</tbody>
</table>

Table 5.10 Average bit rate and PSNR values for “Suzie”, “Salesman” and “Foreman”.

The second set of the simulation results have been illustrated in Figures 5.24-25 and Table 5.11 for the 150-frame “Suzie”, 150-frame “Salesman” and 200-frame “Foreman” sequences. “Suzie”, “Salesman” and “Foreman” were transcoded from 78.118 kbit/s (37.560 dB), 89.700 kbit/s (37.383 dB) and 87.403 kbit/s (33.582 dB) down to 28.908 kbit/s (33.471 dB), 43.630 kbit/s
(34.991 dB) and 46.835 kbit/s (30.029 dB), respectively. The MV refinement window sizes were pre-set as ±4, ±2 and ±5 pixels in the same order as above. The simulations were run at 25 fr/s, with QCIF (176×144 pixels) size and in I-P-P-P-P-... format. These simulation conditions and the transcoding parameters were also valid for the first part of the experiments.

Figure 5.24 Objective results for the combination of AIR and FCS over a C/I = 12 dB CS2 GPRS channel model, a- “Suzie”; b- “Salesman” and c- “Foreman” sequences requiring 3, 4 and 5 timeslots, respectively
Figure 5.25 Subjective results of the 67th (a-f) and 150th (g-j) frames of “Suzie”, 150th (m-p) frames of “Salesman” and 200th frames of “Foreman”, a/h/n/r- error-free; b/i/o/s- error-prone; c/f- FCS only resilient; d/k- AIR only resilient; e/l/p/t- FCS and AIR combined resilient; f/g/m/q- direct enc/dec at higher rate sequences over a C/I = 12 dB CS2 GPRS channel model requiring 3, 4 and 5 timeslots, respectively.
In the second part of the experiments, AIR was also employed for the video transcoding performance tests in addition to the FCS algorithm on a combined platform. This achievement was established to provide the transcoded video streams with the ultimate resilience prior to transmission over fairly high BER GPRS networks. Thus, these particular experiments embrace the tests of a novel combination of the two source coding error resilience algorithms at the video gateway. Consequently, the video gateway is utilised as a remote error-resilient rate management operator within the entire networking scenario. The round-trip delay for the feedback signal was taken as 480 msec (12 frames at 25 fr/s) which included the inherent GPRS round-trip delay of ~450 msec and the additive processing delay times. Each of the simulations were run with 10 different seeds here similar to the first set of the experiments.

<table>
<thead>
<tr>
<th>Type of the Scheme</th>
<th>Av. Bit Rate [kbit/s]</th>
<th>Av. PSNR [dB]</th>
<th>BER</th>
</tr>
</thead>
<tbody>
<tr>
<td>150-frame “Suzie”, MV refinement window size: ±4 pixels, CS2 timeslots: 3</td>
<td>direct enc/dec @ high bit rate</td>
<td>78.118</td>
<td>37.560</td>
</tr>
<tr>
<td></td>
<td>transed error-free</td>
<td>28.908</td>
<td>33.471</td>
</tr>
<tr>
<td></td>
<td>transed non-resilient error-prone</td>
<td>28.908</td>
<td>22.839</td>
</tr>
<tr>
<td></td>
<td>transed FCS only error-resilient</td>
<td>28.796</td>
<td>23.854</td>
</tr>
<tr>
<td></td>
<td>transed AIR only error-resilient</td>
<td>30.504</td>
<td>24.985</td>
</tr>
<tr>
<td></td>
<td>transed FCS+AIR error-resilient</td>
<td>31.377</td>
<td>26.844</td>
</tr>
<tr>
<td>150-frame “Salesman”, MV refinement window size: ±12 pixels, CS2 timeslots: 4</td>
<td>direct enc/dec @ high bit rate</td>
<td>89.700</td>
<td>37.383</td>
</tr>
<tr>
<td></td>
<td>transed error-free</td>
<td>34.916</td>
<td>34.447</td>
</tr>
<tr>
<td></td>
<td>transed non-resilient error-prone</td>
<td>34.916</td>
<td>22.177</td>
</tr>
<tr>
<td></td>
<td>transed FCS+AIR error-resilient</td>
<td>35.227</td>
<td>26.831</td>
</tr>
<tr>
<td>200-frame “Foreman”, MV refinement window size: ±5 pixels, CS2 timeslots: 5</td>
<td>direct enc/dec @ high bit rate</td>
<td>87.403</td>
<td>33.582</td>
</tr>
<tr>
<td></td>
<td>transed error-free</td>
<td>46.835</td>
<td>30.029</td>
</tr>
<tr>
<td></td>
<td>transed non-resilient error-prone</td>
<td>46.835</td>
<td>19.852</td>
</tr>
<tr>
<td></td>
<td>transed FCS+AIR error-resilient</td>
<td>49.725</td>
<td>22.330</td>
</tr>
</tbody>
</table>

Table 5.1 Average bit rate, PSNR and BER values for “Suzie”, “Salesman” and “Foreman” over a C/I = 12 dB CS2 GPRS channel model requiring 3, 4 and 5 timeslots, respectively.

The performance evaluation of the combined AIR and FCS over the GPRS access network employed the CS2 coding scheme at a carrier frequency of 1800 MHz and using the typical urban scenario (TU50) multipath model, where the velocity of the mobile terminal was 50 kph, as specified in [GSM0503] experiments. Furthermore, in both sets of the simulations presented in this sub-section, a 5% of the transmitted video frames were randomly lost. The 5% frame loss case is a typical packet loss rate for GPRS CS2 code at C/I = 12 dB condition [Cella] which was also chosen as the operating point for the error-resilient video transcoding tests over GPRS. Table 5.11 also presents the BERs incurred at C/I = 12 dB using CS2.
5.4.2.4 Analysis of the Results

The 5% frame loss experiments presented varying quality levels with and without the FCS resilience algorithm. This variation was observed to be video sequence-dependent, as depicted in Figures 5.20-22. The effective resilience performance has been shown to rely on the motion activity of the sequence in these figures. Therefore, the performance improvement with the FCS scheme has been demonstrated as ~2.4 dB for the “Suzie” sequence whilst the results of the “Foreman” and “Salesman” sequences showed quality enhancements of ~0.4 dB and ~0.3 dB at most, respectively. This is due to the fact that the particular loss of the very high motion activity frames in the middle of the “Suzie” sequence caused significant quality losses during the simulations, as seen in Figure 5.20 and Table 5.10. Evidently, the FCS scheme performed much better in this particular case as the long-term temporal referencing with feedback signalling achieved an enhancement in the perceptual quality. However, the increase in the round-trip delay of the feedback signal decreased the degree of quality improvement, as observed from Table 5.10. The delay factor simply affected the long-term temporal referencing resulting in more inaccurate references as the back channel signalling time increased.

On the contrary, this experimental observation has been observed to be valid only for “Suzie” when Figures 5.21 and 5.22 are also considered. For “Salesman” and “Foreman”, it has been demonstrated that the quality improvements with the FCS were not as significant as for the “Suzie” sequence. Nevertheless, the resilient transcoding results have also been demonstrated to give better qualities than the non-resilient error-prone transcoding results, as seen in Figures 5.20-22 and Table 5.10. As opposed to the “Suzie” results, these two video sequences presented varying error-resilient transcoding performances with the variation of the feedback signal round-trip time delay. The reason is that the long-term temporal prediction with the FCS algorithm incorrectly referred to the low correlation reference frames (for some certain delay times for the “Foreman” sequence) due to the high motion activity in the “Foreman” sequence. As the waiting time latency increases for the reception of the back channel signal, the lack of correlation between the reference and the current frames causes more intra (I) mode MB transcoding, which increases the output rate whilst also improving the resilient transcoding quality at the same QP value. Conversely, despite its stationary background, fast movements of the hands of “Salesman” also resulted in better correlation in longer time latencies without I-mode transcoding of the MBs.

Moreover, the objective results, presented in the b sides of Figures 5.20-22, and the subjective results, as seen in Figure 5.23, also depict the effects of the round-trip delay on the transcoded video quality. Generally, the obtained results have shown that the transcoding FCS algorithm gives limited improvements on the picture quality compared to when FCS is incorporated within the source coder. This is mainly due to the fact that small MV refinement window sizes put a
limitation on the quality improvement of the motion active scenes in the error-resilient mode. Furthermore, resilience over an already reduced quality video (due to the re-quantisation process at the video transcoder) results in lower performance improvements than source coding resilience techniques.

The rate increase due to the FCS algorithm was also compensated successfully at the video transcoder. Detailed output rate values can be seen in Table 5.10. These results show that in most cases, the bit rate increases as the latency for the feedback signal reception increases. This is due to the lack of correlation between the long-term reference and the current video frames. However, the rate increase can easily be managed with a straightforward adaptive rate reduction algorithm which operates at the resilient video transcoder, as presented here.

The FCS results have proved that even the 12-frame delay resilience cases (480 msec at 25 fr/s) performed well above the non-resilient video communication qualities. Such results have motivated the research to test this particular scheme as a complementary resilience method to the AIR algorithm over the GPRS networks where frame droppings are inevitable. Thus, the results of the tests with the combination of AIR and FCS have been demonstrated in Figures 5.24-25 and Table 5.11. It has been shown that the transcoding with combined resilience achieved superior quality levels against the non-resilient schemes over the GPRS channels with frame losses. The corresponding performance improvements have been presented in these figures as the averages of 4 dB for “Suzie” and “Salesman” and 2.5 dB for “Foreman”, respectively. Moreover, “Suzie” results have been presented in such a way that the quality gains of the combined method of AIR and FCS are compared against the AIR and FCS only resilience results at similar conditions, BER $= 3.3e-03$, over the same GPRS channel at C/I = 12 dB with CS2. The results of these particular experiments have demonstrated 1 dB, 2 dB and 4 dB quality improvements in favour for the FCS only, the AIR only and the combined methods over the non-resilient scheme, respectively. These results can be seen in Figure 5.24.a and Table 5.11. The corresponding quality improvements were achieved with the minimal output bit rate growths with the use of the rate management features of the video transcoder. The bit rate increases due to the use of combined resilience methods were found to be $-2.5$ kbit/s, $-0.3$ kbit/s and $-3$ kbit/s on average for “Suzie”, “Salesman” and “Foreman”, respectively. These increases in bit rates are so little that the GPRS timeslots required for the transmissions of the resilient data and the non-resilient data are exactly the same, which are also mentioned within the figure captions for each of the test sequences individually.

Figure 5.25 presents the subjective results obtained for the combined AIR and FCS tests for the final frames of the 150-frame “Suzie”, 150-frame “Salesman” and the 200-frame “Foreman” sequences. In addition, the 67th frames of the “Suzie” video clip have also been depicted to give a
more lucid perception of the frame loss effects and the improvements obtained with both of the resilience algorithms during transcoding. The reason for choosing the 67th frames of "Suzie" is that these particular frames show the effects of frame losses in a high motion region of the sequence.

5.5 Concluding Remarks

An intermediate stage error resilience addition to an already compressed and transmitted video stream has been discussed in this chapter. For this purpose, a video transcoder has been exploited to produce an error-resilient and standards-compliant output. The resilience was achieved with the use of separate and combined AIR and FCS techniques during the transcoding operations. The trade-off of both resilience schemes, namely the undesired inherent output bit rate increase due to their operations, as discussed in Chapter 2, was easily overcome and resolved by employing an adaptive rate transcoding scheme. Thus, a more efficient adoption of the resilience algorithms could be accomplished with output rates fairly close to the requirements. The adaptive operation of the combined rate and error resilience control feedback loops produced output rates at near target bit rates whilst inserting the necessary amount of robustness to pre-compressed video streams. Numerous experiments gave superior transcoding performances over the error-prone GPRS channels to the non-resilient video qualities.

The AIR transcoding performance has been tested both over random and bursty error-prone channels. These tests have shown that the bursty error effects on the transcoded video were much more detrimental than those of the random errors at similar BER conditions. This is due to the fact that the error-sensitive video data is much more vulnerable to the loss of long bursts of visual information rather than random bit errors. Therefore, interleaving of data prior to its transmission at the video gateway is believed to improve the QoS in error-prone conditions as this will randomise the burstiness of errors. The bursty error-prone channel simulations were carried out using a GPRS mobile-access network model. The inherent GPRS channel interleaving and protection schemes, namely CS1-3, provide a certain degree of protection against transmission errors by means of convolutional coding. However, for video communications, these built-in schemes have been demonstrated to be practically inefficient at higher BER levels. Therefore, the proposal of employing Turbo codes at the video gateway may be quite an effective solution. The simulations have shown that as the protection schemes of the different GPRS channels got weaker, the BER increased significantly. This increase in BER at low C/I ratios, such as C/I = 7, 9 dB, notably degraded the perceptible quality of the video communications. At these low C/I ratios, the resilience provided only by the AIR algorithm was not very satisfactory. This hinted at the
necessity of additional protection/resilience mechanisms, such as FEC, EREC, RVLCs, etc. Conversely, at moderate and high C/I ratios, even a 3-MB AIR method gave quite satisfactory results compared to the non-resilient ones. Despite the addition of AIR to the compressed video data, the transcoder produced video streams which required the same number of GPRS timeslots to be transmitted as the non-resilient ones. For both of the random and bursty error experiments, it has been demonstrated that the detrimental effects of transmission errors and the remedial effects of AIR varied with the change in motion activity within the test sequences. The higher the motion activity, the less robust the video stream to errors.

The FCS transcoding performance has been tested on both the heterogeneous and homogeneous transcoding methods. In the heterogeneous case, the transcoder was designed to be intelligent to inform the MPEG-4 transmitter of the error conditions before the transmission to the H.263 side whilst in the homogeneous one, the video transcoder itself was providing the FCS resilience algorithm. In both cases, video quality has been demonstrated to improve by a couple of dBs. The effect of the round-trip delay for an ACK/NACK has been observed to vary depending on the motion activities in the test sequences. In these tests, it has been shown that the increasing feedback delay also increased the bit rate and affected the video quality due to the lack of correlation between the reference and the current video frames. Similarly, the increase in the bit rate was also easily managed here, as in the AIR experiments, with the rate and error resilience control feedback loops of the video transcoder.

Furthermore, an ultimate combination of the AIR and FCS resilience methods has also been demonstrated. This kind of a combination has been shown to achieve superior transcoding qualities to the non-resilient video qualities at near target output bit rates. Thus, the output streams also required the same number of timeslots as the non-resilient streams. During these particular experiments, 5% video frame loss was also considered in addition to the inherently error-prone GPRS transmission model at C/I = 12 dB and using the CS2 protection scheme. The tests were repeated for several video test sequences and similar results were obtained with 2.5–4 dB quality enhancements in error-prone environments.

In this chapter, the positive effects of the two previously discussed source resilience algorithms have been demonstrated on the transcoded video data. During the numerous experiments, it has been realised that employing AIR and FCS at the video transcoder produced a distinguished performance compared to the source coding resilience applications. The difference occurs only in error-free conditions. The AIR and FCS gradually deteriorate the video qualities when they are applied whilst transcoding in error-free conditions. The reason is that resilience algorithms either refresh or refer to an already de-quantised and re-quantised portion of the video data with already degraded quality due to the quantisation losses. However, the performance of these two resilience
algorithms when applied at the transcoding stage appears to be superior in error-prone conditions, as demonstrated with a vast number of simulation results.

Since this chapter has presented an incorporation of the error resilience schemes into the video transcoding algorithm, it consequently shows another objective of the video transcoders: the provision of error resilience to compressed video streams. Thus, it can be said that the video gateway can carry most of the burden of the networks allowing the source encoders and end-decoders to stay free of complex resilience and/or rate regulation tasks in the future.

The novelties presented in this chapter can be summarised as:

* The design of a novel adaptive error-resilient video transcoding mechanism with a near target output bit rate performance for reliable low bit rate video communications over mobile-wireless channels.

* The first time employment of the temporal resilience algorithms, namely AIR and FCS, at the video transcoding stage for a pre-compressed video stream.

* A comprehensive study of the two resilient transcoding algorithms over a model of the GPRS network.

* The novel integration of the two resilience schemes to combat the quality degrading error characteristics and the frame loss natures of wireless packet networks.
Chapter 6

6 Conclusion

6.1 Preamble

Interoperability of diverse networks is promised to be supported by UMTS in very near future. UMTS represents a new generation of mobile communication system in a world where personal services are based on a combination of fixed and mobile radio services providing a seamless end-to-end service to users. This global and universal mobile communication infrastructure will give a support to transmission bit rates of up to 2.048 Mbit/s which will offer unified services to users in wireless and circuit-switched environments for a very wide range of mobile multimedia communications, services and applications. UMTS services have been planned to support up to 384 kbit/s outdoors and up to 2 Mbit/s indoors. From this point of view, MPEG-4, which is designed to provide efficient mobile multimedia-oriented services and applications in a broad range of transfer rates from very low bit rates (few kilobits per second) up to very high bit rates (tens of megabits per second), will play a very important role in the entire UMTS network. Consequently, traffic and network asymmetries will draw much of the attention as well as the significant amount of efforts. Thus, today's PSTN video standard H.263 will require an essential link to the universal mobile telecommunication system for a global interconnection. In this work, the proposed solution has appeared to be the transparent connection of H.263 and MPEG-4, which are genuinely believed to be the most adequate video coding standards for UMTS communications, via a video transcoder yet in an error-resilient way.

Video transcoding is essential for an end-to-end compatibility of different homogeneous or heterogeneous multimedia networks. MPEG-4 was originally designed to operate on different media, including PSTNs, satellite networks and particularly the mobile-wireless channels. However, H.263 has been operating on those circuit-switched networks quite successfully for a considerable amount of time. Therefore, it is a very momentous issue to efficiently interconnect these two video standards, running on different network topologies. This can be achieved by introducing a video transcoder at the interconnection points of different networks where standards meet each other, such as at an MCU. The main idea is to achieve necessary syntax conversions and translations by mapping the video standards to each other so as to enable the client receiver to
decode whatever was encoded previously by the other standard. Besides the standard conversions, transmission (bit and frame) rate and/or resolution conversions can also be accomplished with the use of same video transcoder. Thus, matching of input network characteristics and constraints to output network characteristics and constraints in a seamless way can be achieved.

In the light of these facts, several video transcoding algorithms have been investigated in this research work. These transcoding algorithms have achieved efficient interconnection of diverse video communication networks. In the beginning, the priority has been given to the congestion resolutions of various networks with varying bandwidth requirements. Therefore, homogeneous transcoding algorithms have been presented to perform the essential transmission rate management operations. Following, the focus has been placed on the interoperability of the two different low bit rate video compression standards, namely H.263 and MPEG-4. For this purpose, heterogeneous video transcoding algorithms have been introduced to achieve the essential interconnection between the two standards without the need for unnecessary complexities and processing delays. Finally, in order to render the previously derived two video transcoding algorithms robust to channel error impacts, an ultimate error-resilient video transcoding scheme has been designed. Numerous tests on the efficacy of the error-resistant transcoder has shown that it is indeed possible to mitigate the error effects at a centralised location within the network to share the complex resilience and rate regulation burdens of source encoders and decoders.

In this chapter, all the former concluding remarks are summarised in the second section for a final conclusive opinion on the overall research work. Lastly, the third section presents the thoughts of future work whilst taking this thesis as a basis for prospective ideas.

### 6.2 Concluding Overview

The aim of the research program presented in this thesis was to establish the basics for a generic video transcoder. The solutions have been proposed to alleviate several interoperability problems, such as diverse congestion or bandwidth limitations, varying QoS properties due to channel errors and divergent syntax requirements imposed by different video compression schemes. In this sense, the thesis has examined each problematic issue individually whilst also providing the proposed remedial solutions. The discussions have started with a preliminary background knowledge and led the readers to more elaborate methods and designs. In this chapter, a re-collection of those algorithms are given.

Chapter 1 has formed the introductory chapter to the thesis. In this chapter, the objectives of the research work have been defined. The performance assessment techniques used throughout the
thesis have been introduced. Furthermore, the original contributions have also been outlined in several bullet-points.

Chapter 2 has primarily comprised the background chapter of the thesis. Thus, this chapter has gathered several research work-related topics for a clearer understanding of the rest of the thesis. In this chapter, an answer to the question “where and when low bit rate video coding” has been sought. Therefore, low bit rate video coding and compression have been discussed in many aspects, such as user and network perspectives, applications, requirements and techniques. The profiles of two low bit rate video coding standards, namely H.263 and MPEG-4, have briefly been outlined. These standards have been examined for their operations with and without two motion-related operational modes, namely advanced prediction and unrestricted MV modes. The tests on these two modes have demonstrated that both video coding standards achieve better performance in high motion active scenes in the case that they are switched on. The enhancement in the video quality increases with the increasing motion activity. Moreover, these tests have also shown that MPEG-4 video quality is slightly better than that of H.263 video at similar compression ratios and transmission bit rates. In addition, the source material used for the performance assessments during the research work have also been introduced in this chapter.

In addition, the importance of error resilience in video communications has also been highlighted in Chapter 2. For this particular purpose, channel error effects on received video quality have been examined. It has been discussed that channel errors severely degrade the QoS of video communications as video compression schemes highly depend on VLCs and predictive coding techniques. These two methods remove the unnecessary redundancies from video signals resulting in high compression efficiencies whilst rendering the resulting bitstream critically susceptible to error impacts. Thus, these impacts severely deteriorate the transmitted video quality due to error propagation. Therefore, this chapter has also discussed a remedial solution, error resilience, to achieve robust video transmissions. Amongst several methods, two of the resilience algorithms, AIR and FCS, have been described in detail as they were selected to mitigate bit error and/or frame loss effects in the course of the research work due to their low complexity and standards-compliant operations. AIR is a method which regularly refreshes the high motion region of interest of a video sequence whereby FCS imposes an adaptation of the coding method on the source coder to frame loss cases due to congestion and/or severe error-prone conditions. Tests have demonstrated notable quality enhancing properties of both error resilience algorithms yet at the expense of increased output bit rates.

Chapter 3 has described what video transcoding is and why it is needed. In this part, three different categories of video transcoding algorithms have been introduced: homogeneous, heterogeneous and error-resilient video transcoding. These three categories have then comprised
the major discussions of the three core chapters of this thesis, namely Chapter 3, Chapter 4 and Chapter 5.

Thus, Chapter 3 has been dedicated to the fully comprehensive study of several existing and non-existing proposed homogeneous video transcoding techniques which form the first category. In this chapter, the transcoding method has further been divided into three, in which bit rate, frame rate and resolution reductions have comprised the main concerns. Furthermore, bit rate reduction schemes have also been analysed in five different methods of tandem. Here, the discussions have initiated with a basic method of tandem, namely cascaded fully decoding and re-encoding method, which has been led to more efficient and less complex transcoding algorithms, namely open-loop and closed-loop algorithms. The performance tests carried out using different tandem operations have demonstrated that neither the cascaded method nor any of the transcoding algorithms give as good quality as the direct encode/decode operation at a given bit rate. The quality is always slightly lower due to the re-quantisation process of DCT coefficients. However, amongst each other, the drift-free schemes have performed equally satisfactory in terms of QoS regarding highly asymmetrical bit rate conversions (reductions). Open-loop transcoding method has presented the worst qualities in all experiments as the accumulative drift effect significantly deteriorates the video quality in time. This quality loss has been reported to be of up to 12 dB on average in the simulated test sequences. On the other hand, closed-loop techniques have been demonstrated to bear complexity increase versus quality improvement trade-off in high motion active scenes. Consequently, a compromise has been found between low complexity and good motion quality with the design of an MV refinement algorithm within the transcoding operation. With this method, in very small search windows, it has been observed that the quality of transcoding of high motion images increases with a notably low complexity due to the refinement of input non-optimal MVs. The method works extremely well for high transcoding asymmetries where there is a big difference in the input and output video bit rates. Moreover, the MV refined transcoding quality also improves as the motion activity increases in the transoded video scenes. The tests have also shown that the quality increases with the enlarging small MV refinement window sizes at the cost of increased bit rates and complexities. To make use of this particular feature, an automatic self-adjusting MV refinement window has been designed which traces the motion activity in a video scene. This window expands for high motion parts and shrinks for low motion ones. It has been demonstrated that the automatic scheme performs good quality transcoding operations with a mediation amongst QoS in high motion activity, complexity and targeted transcoding rate.

Furthermore, this chapter has also presented an algorithm for frame rate reductions. The designed method intentionally drops video frames to meet the transmission rate reduction requirements.
This is an extremely momentous operation that at times when bit rate reduction is not enough solely, additional frame rate reduction gives user the flexibility of receiving the video data even at very low transmission rates. Moreover, frame rate reduction is accomplished to more efficiently allocate the remaining bits of a video sequence for enhanced qualities. During the experiments, it has been shown that the design of an MV logbook, which keeps a track of motion data of the intentionally dropped frames, gives a better transcoding quality. This is due to the fact that transcoded video frames also carry the necessary motion information that would conventionally be missing for the dropped frames.

Lastly, this chapter has also conferred a proposed multimedia traffic planning system in which a set of video transcoders operate in parallel forming a video transcoder bank. The bank of transcoders has been shown to perform the necessary bit rate management whilst also catering for varying congestion conditions. The adaptive operation of the video transcoder bank has given improved overall video transcoding qualities even at long congestion periods.

Chapter 4 has elaborated the second category of video transcoding algorithms: heterogeneous video transcoding. In this chapter, the interoperability between MPEG-4 and H.263 has comprised the major networking concerns. Thus, the similarities and the discrepancies between the two low bit rate video coding standards have been carefully investigated in this chapter. As a consequence, this study has led the research to the design of an MPEG-4/H.263 bi-directional heterogeneous video transcoder, bearing the future 3G networking requirements in mind. The experimental results have demonstrated that the transcoding quality is superior over the conventional tandem process, namely cascaded fully decoding/re-encoding, with an extremely low complexity operation. This is due to the fact that the designed transcoding algorithm works totally in compressed domain whereas the cascaded method requires an extra decode and re-encode cycle rendering video data reduced qualities.

Finally, Chapter 5 has presented the third category transcoding operation: error-resilient video transcoding algorithm. In this chapter, video transcoding architecture has been designed such that it performs necessary rate management operations whilst also providing error resilience to video streams. This is a significant achievement since the error resilience addition is accomplished to an already compressed non-resilient video data at an intermediate stage within an entire network. The system operates with two resilience algorithms, namely AIR and FCS. AIR provides error resilience by stopping the accumulative effects of bit errors over error-prone channels whilst FCS gives robustness against the detrimental effects of frame losses due to heavy traffic (congestion) and/or error conditions. Both algorithms have demonstrated to give significantly improved qualities under severe error conditions (i.e. BER =1e-02/-03). Moreover, the associated output rate increase, as described in Chapter 2, has successfully been compensated with the rate management
features of the designed transcoder. As a result, enhanced video qualities over error-prone channels have been achieved with near target transmission rates. The algorithm has also been tested on GPRS mobile-access networks and similar good quality results have been obtained. Thus, this chapter has also discussed the GPRS networking structure. In the computer simulation experiments, it has been demonstrated that GPRS channels can severely damage the QoS of video communications depending on the operating points and selected coding schemes. For instance, using CS3 at $C/I = 7, 9$ dB, video qualities have been extremely damaged. This has also affected the resilient video qualities. However, higher $C/I$ conditions and better CSs have resulted in increased qualities. This is due to the fact that the higher the $C/I$ and the better the CS, the lower the BER. With the rate management skills of the error-resilient transcoder, the output streams only require as many GPRS timeslots as conventionally needed by the non-resilient transmissions. Thus, it has been demonstrated that error resilience can later be added to an already compressed video stream without any significant rate increase at a centralised position in a network.

Lastly, the FCS and AIR algorithms have been combined on a common platform for the design of an ultimate error-resilient video transcoder. Numerous tests on the performance of such a design have demonstrated superior transcoding qualities in real-life GPRS networking scenarios where BER $\equiv 1e^{-03}$ and round-trip delays exceed $\sim 450$ msec including additional processing delays.

Table 6.1 presents a more compact summary of the individual conclusions obtained from each of the chapters of this thesis.

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Conclusions</th>
</tr>
</thead>
</table>
| **Chapter 1** | objectives defined  
| | performance evaluation methods introduced  
| | original contributions outlined  |
| **Chapter 2** | background information given  
| | the importance of low bit rate video communications discussed  
| | H.263 & MPEG-4 described  
| | error resilience necessity highlighted  
| | source material used for the experiments introduced  
| | tests carried out related to pure video coding & error resilience algorithms (AIR & FCS)  
| | test results presented improved service qualities with the addition of resilience  |
| **Chapter 3** | video transcoding necessity & three major types briefly overviewed  
| | the first type of video transcoding method, namely homogeneous video transcoding algorithm, detailed  
| | bit rate reduction mechanisms described & tested  
| | frame rate reduction algorithm described & tested  
| | resolution reduction technique discussed  
| | an adaptive multimedia traffic planning algorithm with the use of video transcoders described & tested  
| | test results demonstrated good service quality levels for transmission rate regulation operations  |
Chapter 4
- the second type of video transcoding method, namely heterogeneous video transcoding algorithm, detailed
- the necessity of interoperability between heterogeneous network topologies highlighted
- an efficient MPEG-4/H.263 video transcoder designed & tested
- test results showed improved service quality levels with highly reduced complexity

Chapter 5
- the third type of video transcoding method, namely error-resilient video transcoding algorithm, detailed
- the addition of error resilience to an already compressed non-resilient video stream at a centralised stage discussed
- an adaptive error-resilient video transcoder architecture described in detailed
- error resilience provided by AIR, FCS & combinations of the two
- the described system tested over several real-life scenarios, such as random error channels & GPRS network
- test results demonstrated better service qualities with error-resilient transcoding at near target output bit rates.

Chapter 6
- overall conclusions given
- concluding remarks for each of the previous chapters summarised
- thoughts on future work expressed

Table 6.1 The summary of concluding remarks of thesis chapters for a quick reference

6.3 Thoughts on Future Work

The heterogeneity of diverse networks and user requirements will play an important role in the universal interoperability issues in the very near future. Fast developing multimedia era will force the interoperability issues to be resolved in an optimised and rapid way quite soon. The solution seems to be the intelligent video gateways comprising efficient video transcoders. Therefore, this research work has been devoted to this potentially significant scientific problem and a few thoughts to carry out a prospective study in the same curious manner are discussed in this section. The published papers and the encouraging feedback received from numerous researchers working in the similar fields have already proved that the selected research area is still wide open to new studies and contributions. Thus, a few promising possible research areas that have been thought to bear significance are as follows:

Firstly, the error-resilient video transcoder architecture that has been designed in this work can be made more robust to varying channel error conditions. The encouraging success of the combined AIR and FCS methods at the transcoding level can lead the future research work to include more efficient resilience algorithms operating in harmony. A number of the algorithms considered have already been mentioned in the relevant chapter. EREC is a very suitable resilience method since it does not incur any overheads whilst providing self-synchronisation of video data. Addition of re-synchronisation methods together with two-way decoding system using RVLCs are another way of increasing the error robustness. Moreover, unequal data prioritisation can achieve good
transcoding qualities where high priority video information, such as headers and MVs, is highly protected or transmitted with high power or over a low BER channel. Thus, multi-threading operations can also be utilised within the transcoder to decide which portion of the video data will be sent via what type of a transmission medium simultaneously. In [Dogan1], the combination of different resilience techniques has been described for source coding mechanisms. With the described transcoding operation, the error resilience addition has moved to a more centralised point within the network. This enables a flexible error resilience inclusion where and when it is required. Furthermore, video transcoder can also be enabled to perform concealment techniques whereby the received video signal is already erroneous.

Secondly, for an ultimate interconnection establishment, the homogeneous and the heterogeneous video transcoding algorithms, which have been described, have to be linked to each other. Thus, the resulting video transcoder will not only achieve bandwidth or congestion-related transmission rate reductions, but it will also alleviate standard and syntax impairments between MPEG-4 and H.263. Moreover, the addition of error resilience capabilities increases the transcoder flexibility whilst combining the three categories of transcoding algorithms for the first time in literature. However, having successfully achieved the MPEG-4 and H.263 interconnection, the focus has now to be placed on different syntax conversions, such as very much demanding MPEG-2/MPEG-4 and H.26L/MPEG-4 translations. This is particularly a momentous issue as very high rate MPEG-2 broadcast standard will need to be accessed by very low rate MPEG-4 mobile devices. The increasing number of MPEG-2/H.263 video standard conversions found in the literature, [Feams] and [Shana3], proves that the multimedia networking trend is moving towards very high to low rate transcoding requirements.

Thirdly, the research work presented in this thesis was mainly based on low bit rate mobile video communications for future 3G networks. Therefore, resolution reductions of the input video signals have been discussed but not analysed as QCIF size video has been accepted to be suitable for such communications. Furthermore, due to low bit rate and low complexity mobile communication requirements, transcoding algorithms have been performed in simple profile video whereby object-based coding techniques have not been involved. However, future heterogeneous communication environments will also comprise desktop multimedia equipment. Thus, apart from the resolution reduction needs, segmentation based object-oriented encoding and transcoding will also be quite effective in the future. Particularly having considered the very high error sensitivity of video object shape data, resilient transcoding methods will be of extreme importance. In addition, video transcoders can also be designed to integrate graphics into an already compressed video signal where necessary.
Next, the video transcoding has been analysed in many aspects in this thesis to constitute an ultimate video gateway design. Another thought on the future research work is the design of a user gateway which interacts with the already designed video gateway. The user gateway is a device located at the user end which is aware of the recipient capabilities. Therefore, a more comprehensive networking scenario can be established with the combined operation of a video gateway (at a centralised point) and a user gateway (at the receiving end point). In this way, the video gateway is enabled to receive necessary feedback information from the end-user. This return signal can inform the video gateway of channel congestion or error conditions, user bandwidth restrictions or power limitations, display constraints and specific demands/needs as well as the user’s interactivity requirements. Thus, more intelligent video transcoding and transmission operations can be achieved to effectively suit the user- and network-driven profiles. The overall system gives a chance to form a more personalised networking infrastructure.

Lastly, the video gateway idea can be taken as the basis for an advanced state-of-the-art multimedia gateway model for the global multimedia data. The new system can be designed to perform similar operations to video transcoding but on other multimedia information, such as speech, audio and data. In this way, a future-proof, application- and standards-independent interoperability can be achieved amongst numerous multimedia networks. The prospective scenario which comprises three kinds of gateways, namely video, user and multimedia, are illustrated in Figure 6.1.
Appendix A

List of Publications


- S. Dogan, A.H. Sadka and A.M. Kondoz, "Tandeming/transcoding issues between MPEG-4 and H.263", Mobile and Personal Satellite Communications 3, Proceedings of the 3rd European Workshop on Mobile/Personal Satcoms, EMPS'98,
Appendix B

List of Abbreviations

3G third generation
AC high order frequency coefficient
ACK acknowledgement
ACTS advanced communication technology and services
AIR adaptive intra refresh
ARQ automatic repeat request
ATLANTIC advanced television at low bit rates and networked transmission over integrated communication system
ATM asynchronous transfer mode
AVO audio-visual object
B bi-directional
BER bit-error-rate
B-ISDN broadband-integrated services digital network
BS base station
C/I carrier-to-interference ratio
Ch, Cr chrominance components
CCSR Centre for Communication Systems Research
CD committee draft
CDROM compact disc read-only memory
CIF common intermediate format
CS coding scheme
DC lowest order frequency coefficient (0 Hz)
DCT discrete cosine transform
DFDS displaced frame difference signal
DIS draft international standard
DVB digital video broadcasting
DVD digital versatile disk
EDGE enhanced data rates for global system for mobile telecommunication evolution
EREC error-resilient entropy coding
ETSI European Telecommunication Standards Institute
FCS feedback control signalling
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FEC</td>
<td>forward error correction</td>
</tr>
<tr>
<td>FLC</td>
<td>fixed length codeword</td>
</tr>
<tr>
<td>GPRS</td>
<td>general packet radio service</td>
</tr>
<tr>
<td>GSM</td>
<td>global system for mobile telecommunications</td>
</tr>
<tr>
<td>HDTV</td>
<td>high definition television</td>
</tr>
<tr>
<td>HVS</td>
<td>human visual system</td>
</tr>
<tr>
<td>I</td>
<td>intra</td>
</tr>
<tr>
<td>IDCT</td>
<td>inverse discrete cosine transform</td>
</tr>
<tr>
<td>IP</td>
<td>Internet protocol</td>
</tr>
<tr>
<td>IS</td>
<td>international standard</td>
</tr>
<tr>
<td>ISDN</td>
<td>integrated services digital network</td>
</tr>
<tr>
<td>ISO</td>
<td>International Standards Organisation</td>
</tr>
<tr>
<td>ITU</td>
<td>International Telecommunication Union</td>
</tr>
<tr>
<td>JPEG</td>
<td>Joint Picture Experts Group</td>
</tr>
<tr>
<td>LVQ</td>
<td>lattice vector quantiser</td>
</tr>
<tr>
<td>MB</td>
<td>macroblock</td>
</tr>
<tr>
<td>MCU</td>
<td>multipoint control unit</td>
</tr>
<tr>
<td>M-JPEG</td>
<td>motion-Joint Picture Experts Group</td>
</tr>
<tr>
<td>MoMuSys</td>
<td>Mobile Multimedia Systems</td>
</tr>
<tr>
<td>MPEG</td>
<td>Motion Picture Experts Group</td>
</tr>
<tr>
<td>MSC</td>
<td>mobile switching centre</td>
</tr>
<tr>
<td>MSE</td>
<td>mean-square-error</td>
</tr>
<tr>
<td>MV</td>
<td>motion vector</td>
</tr>
<tr>
<td>NACK</td>
<td>non-acknowledgement</td>
</tr>
<tr>
<td>P</td>
<td>inter, predictive</td>
</tr>
<tr>
<td>PDCH</td>
<td>packet data channel</td>
</tr>
<tr>
<td>PLR</td>
<td>packet-loss-ratio</td>
</tr>
<tr>
<td>PSNR</td>
<td>peak-to-peak signal-to-noise ratio</td>
</tr>
<tr>
<td>PSTN</td>
<td>public-switched telephone network</td>
</tr>
<tr>
<td>QCIF</td>
<td>quarter common intermediate format</td>
</tr>
<tr>
<td>QoS</td>
<td>quality of service</td>
</tr>
<tr>
<td>QP</td>
<td>quantisation parameter</td>
</tr>
<tr>
<td>RVLC</td>
<td>reversible variable length codeword</td>
</tr>
<tr>
<td>S</td>
<td>sprite</td>
</tr>
<tr>
<td>SAD</td>
<td>sum of absolute difference</td>
</tr>
<tr>
<td>SAD_th</td>
<td>sum of absolute difference threshold</td>
</tr>
<tr>
<td>SDQE</td>
<td>sum of differential quantisation error</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>TETRA</td>
<td>terrestrial trunked radio system</td>
</tr>
<tr>
<td>TFI</td>
<td>temporary flow identifier</td>
</tr>
<tr>
<td>TMN</td>
<td>Test Model</td>
</tr>
<tr>
<td>TU</td>
<td>typical urban</td>
</tr>
<tr>
<td>UMTS</td>
<td>universal mobile telecommunication service</td>
</tr>
<tr>
<td>VLC</td>
<td>variable length coding/codeword</td>
</tr>
<tr>
<td>VLSI</td>
<td>very large scale integration</td>
</tr>
<tr>
<td>VM</td>
<td>Verification Model</td>
</tr>
<tr>
<td>VO</td>
<td>video object</td>
</tr>
<tr>
<td>VoD</td>
<td>video-on-demand</td>
</tr>
<tr>
<td>VOL</td>
<td>video object layer</td>
</tr>
<tr>
<td>VOP</td>
<td>video object plane</td>
</tr>
<tr>
<td>VSAT</td>
<td>very small aperture terminal</td>
</tr>
<tr>
<td>WD</td>
<td>working draft</td>
</tr>
<tr>
<td>Y</td>
<td>luminance component</td>
</tr>
</tbody>
</table>
Bibliography


Bibliography


[GSM0503] ETSI/SMG GSM 05.03: “Digital cellular telecommunications system (GSM radio access phase 3); channel coding”, Version 6.1.

[GSM0505] ETSI/SMG GSM 05.05: “Digital cellular telecommunications system (GSM radio access phase 2+); radio transmission and reception”, Version 7.0.


