

Spatio-temporal pre-processing methods for region-of-interest video coding

Linda S. Karlsson



Mittuniversitetet
MID SWEDEN UNIVERSITY

Department of Information Technology and Media
Mid Sweden University

Licenciate Thesis No. 21
Sundsvall, Sweden
2007

ISBN 978-91-85317-45-5
ISSN 1652-8948

Mittuniversitetet
Informationsteknologi och medier
SE-851 70 Sundsvall
SWEDEN

Akademisk avhandling som med tillstånd av Mittuniversitetet framlägges till offentlig granskning för avläggande av teknologie licenciatexamen fredagen den 27 april 2007, 13.00 - 15.00, i L111, Mittuniversitetet, Holmgatan 10, Sundsvall.

©Linda S. Karlsson, april 2007

Tryck: Tryckeriet Mittuniversitetet

To My Family

Abstract

In video transmission at low bit rates the challenge is to compress the video with a minimal reduction of the perceived quality. The compression can be adapted to knowledge of which regions in the video sequence are of most interest to the viewer. Region of interest (ROI) video coding uses this information to control the allocation of bits to the background and the ROI. The aim is to increase the quality in the ROI at the expense of the quality in the background. In order for this to occur the typical content of an ROI for a particular application is firstly determined and the actual detection is performed based on this information. The allocation of bits can then be controlled based on the result of the detection.

In this licenciate thesis existing methods to control bit allocation in ROI video coding are investigated. In particular pre-processing methods that are applied independently of the codec or standard. This makes it possible to apply the method directly to the video sequence without modifications to the codec. Three filters are proposed in this thesis based on previous approaches. The spatial filter that only modifies the background within a single frame and the temporal filter that uses information from the previous frame. These two filters are also combined into a spatio-temporal filter. The abilities of these filters to reduce the number of bits necessary to encode the background and to successfully re-allocate these to the ROI are investigated. In addition the computational complexities of the algorithms are analysed.

The theoretical analysis is verified by quantitative tests. These include measuring the quality using both the PSNR of the ROI and the border of the background, as well as subjective tests with human test subjects and an analysis of motion vector statistics. The qualitative analysis shows that the spatio-temporal filter has a better coding efficiency than the other filters and it successfully re-allocates the bits from the foreground to the background. The spatio-temporal filter gives an improvement in $PSNR_{ROI, Avg}$ of more than 1.32 dB or a reduction in bitrate of 31 % compared to the encoding of the original sequence. This result is similar to or slightly better than the spatial filter. However, the spatio-temporal filter has a better performance, since its computational complexity is lower than that of the spatial filter.

Acknowledgements

Firstly I would like to thank my supervisors Mårten Sjöström, Youzhi Xu and Tingting Zhang for providing the necessary help and guidance making this licenciate thesis possible. In particular Mårten Sjöström for the help with writing and for the questions that has helped to make it possible for others than me to understand my writing. I would also like to express my gratitude to Roger Olsson for all the valuable comments and discussion of my work and for the interesting discussions of other parts of life. I would also like thank the others envolved in the project "Framtidens distribution av multimedia" for their input and discussions. In particular Ulf Jennehag for the discussions on various topics in video source coding, which has contributed to my understanding of the area, and Stefan Pettersson for reading and commenting the thesis. In addition I would like to thank Karl W Sandberg and Rolf Dahlin for the advice concerning tests with human test subjects.

Many thanks the other members of the Information and Communication System Division at Mid Sweden University for providing a friendly and welcoming environment. All the interesting discussions of various nature and yatzy playing during breaks have helped me stay motivated. In particular I would like to acknowledge Daniel Forsgren, Mikael Gidlund, Ulf Reiman, Chinghua Wang, Patric Österberg for making my time at the fifth floor enjoyable.

I would also like to express my gratitude to the Swedish Graduate School in Telecommunications for providing interesting summer schools, qualified courses and financial support. In addition I would like to thank the EU Objective 1-programme Södra Skogslän region for funding my research.

Last but not least I would like to give my thanks to my family and friends that have supported and encouraged me all the way and for making sure that my life does not only consist of research and studies.

Sundsvall 2007-03-29

Linda Karlsson

Contents

Abstract	v
Acknowledgements	vii
List of Papers	xi
Terminology	xiii
1 Introduction	1
1.1 Video coding	2
1.1.1 Block-based hybrid coding	2
1.1.2 Video coding standards	4
1.2 Perceptual quality	5
1.3 Region-of-interest video coding	5
1.3.1 Applications	5
1.3.2 Foveated coding	7
1.3.3 Object-based coding	8
1.4 Overall aim	8
1.5 Scope	8
1.6 Outline	9
1.7 Contributions	9
2 ROI video coding	11
2.1 ROI detection	11

2.1.1	Visual attention	11
2.1.2	Face detection	12
2.2	Bit allocation	14
2.2.1	Spatial bit allocation	14
2.2.2	Temporal bit allocation	15
2.2.3	Combinations of spatial and temporal bit allocation	15
2.3	Other applications of ROI	16
2.4	Quality measures	16
2.4.1	Objective quality	16
2.4.2	Subjective quality	17
3	Spatial filtering	19
3.1	Block-based hybrid coding of a video sequence	19
3.2	The SP filter algorithm	24
3.2.1	ROI detection	25
3.2.2	Quality map	25
3.2.3	SP filter	26
3.3	Rate-Distortion of SP filtered video	27
3.3.1	Intra-coded frames	27
3.3.2	Inter-coded frames	28
3.4	Computational complexity	30
3.4.1	Quality map	31
3.4.2	Spatial filter	31
3.5	Experimental setup	34
3.6	Experimental results	37
3.6.1	SP filtering using several Gaussian filters	37
3.6.2	Reduction of computational complexity	42
3.6.3	Subjective tests	45
3.7	Chapter summary	45

4	Temporal filtering	49
4.1	Temporal filter	49
4.2	Rate-Distortion of TP filtered video	53
4.2.1	Intra-coded frames	53
4.2.2	Inter-coded frames	54
4.3	Computational complexity	57
4.4	Experimental setup	58
4.5	Experimental results	58
4.5.1	Bitrate	59
4.5.2	PSNR of the ROI and the transition region	60
4.6	Chapter summary	61
5	Spatio-temporal filtering	65
5.1	The SP filter	65
5.2	The TP filter	65
5.3	Coding efficiency of the background and reallocation	68
5.4	Computational complexity	69
5.5	Experimental setup	69
5.6	Experimental results	70
5.6.1	Bitrate	70
5.6.2	PSNR	72
5.7	Chapter summary	73
6	Qualitative and quantitative comparison of the filters	75
6.1	Comparison of qualitative tests.	75
6.1.1	Computational complexity	77
6.2	Experimental setup	77
6.2.1	Motion vector analysis	78
6.2.2	Subjective tests	78
6.3	Experimental results	80

6.3.1	Bit rate	81
6.3.2	PSNR	81
6.3.3	Motion vector analysis	83
6.3.4	Subjective tests	88
6.4	Chapter summary	93
7	Conclusions	95
7.1	Summary and discussion	95
7.2	Future works	98
	Bibliography	101
A	Parametric skin detection model	107
B	$PSNR_{Border, Avg}$ for SP filtering	109
C	The α parameter	111
	Biography	113

List of papers

This thesis is based on the following papers:

- I L. S. Karlsson and M. Sjöström. A preprocessing approach to ROI video coding using variable Gaussian filters and variance in intensity. In *47th International Symposium ELMAR-05 focused on Multimedia Systems and Applications, Zadar, Croatia*, pp. 65-68, 2005.
- II L. S. Karlsson and M. Sjöström. Improved ROI video coding using variable Gaussian pre-filters and variance in intensity. In *2005 IEEE International conference on Image Processing, ICIP, Genua, Italy*, Vol. 2, pp. 313-316, 2005.
- III L. S. Karlsson, R. Olsson and M. Sjöström. Temporal filter with bilinear interpolation for ROI video coding. Technical report, MUCOM, Department of Information Technology and Media, Mid Sweden University, Sundsvall, Sweden, 2006.
- IV L. S. Karlsson, M. Sjöström and R. Olsson. Spatio-temporal filter for ROI video coding In *14th European Signal Processing Conference, EUSIPCO, Florence, Italy*, 2006

Terminology

Abbreviations and Acronyms

B	8x8 pixels Block
Bg	Background
CABAC	Context-based Adaptive Binary Arithmetic Coding
DCT	Discrete Cosine Transform
fps	Frames Per Second
HSI	Hue, Saturation, Intensity
HSL	Hue, Saturation, Luminance
HSV	Hue, Saturation, Volume
HVS	Human Visual System
ITU	International Telecommunication Union
kbps	Kilobits Per Second
MB	16x16 pixels Makroblock
MC	Motion Compensation
MPEG	Moving Photographic Experts Group
ME	Motion Estimation
MSE	Mean Square Error
MVD	Motion Vector Difference
LUT	Lookup Table
PFM	Previous Frame Memory
PSNR	Peak Signal to Noise Ratio
QCIF	Quarter Common Intermediate Format containing video sequences of 30 fps and 176×144 pixels per frame.
QUANT	Quantization
RGB	Red, Green, Blue.
ROI	Region Of Interest
SAD	The Sum of Absolute Differences.

SOM	Self-Organizing Map
SP	Spatial
TP	Temporal
SPTP	Spatio-temporal
TSL	Tint, Saturation, Luminance
VLC	Variable Length Code
YCbCr	Colorspace with one luma component and two chrominance components, blue and red
CbCr	The chrominance plane of the YCbCr color space.

Mathematical Notation

A_{Bg}	The minimum value of Q included in the transition region.
A_{ROI}	The minimum value of Q included in the ROI.
B_{DM}	Binary detection map.
$B_{TP}^{(p,q)}$	Block of size 8×8 and index (p,q) , where $p = \lceil m/8 \rceil$ and $q = \lceil n/8 \rceil$.
$B_V^{(u,v)}$	Block of size $U \times U$ and index (u,v) , where $u = \lceil m/U \rceil$ and $v = \lceil n/U \rceil$.
$C_b^{(f,(m,n))}$	The value of the Cb component of $I^{(f,(m,n))}$ for pixel (m,n) in frame f .
$C_r^{(f,(m,n))}$	The value of the Cr component of $I^{(f,(m,n))}$ for pixel (m,n) in frame f .
C	The region $C \in \{Bg, ROI\}$.
D	Distortion of the sequence introduced by the compression.
$D^{(f)}$	Distortion of frame f .
$D_{Bg}^{(f)}$	Distortion of the background of frame f .
$D_{Bg,SP}^{(f)}$	Distortion of the background of the spatially filtered frame f .
$D_C^{(f)}$	Distortion of the region C of frame f .
$D_{C,Filter}^{(f)}$	Distortion of the region C of the filtered frame f .
\bar{d}_{MV}	Motion vector.
$\bar{d}_{MV}^{(f-1)}$	The motion vector in frame f in the original case that point to the block in $f-1$, whose motion vector $\bar{d}_{MV}^{(f-2)}$ points at the same block in $f-1$ as $\bar{d}_{MV,TP}$ in the filtered case.
$\bar{d}_{MV}^{(f,(k,l),e)}$	Motion vector e in MB (k,l) of frame f .
$d_{MV,1}^{(f,(k,l),e)}$	Motion vector component 1 for motion vector e in MB (k,l) of frame f .

$d_{C,MV,1}^{(f,(k,l),e)}$	Motion vector component 1 for motion vector e in MB (k,l) of region C in frame f .
$\bar{d}_{MV}^{(f-2)}$	The motion vector in the original case, which points at the same block in $f - 1$ as $\bar{d}_{MV,TP}$ in the filtered case.
$d_{MV,2}^{(f,(k,l),e)}$	Motion vector component 2 for motion vector e in MB (k,l) of frame f .
$d_{C,MV,2}^{(f,(k,l),e)}$	Motion vector component 2 for motion vector e in MB (k,l) of region C in frame f .
$\bar{d}_{MV,TP}$	Motion vector in TP filtered sequence.
$D_{ROI}^{(f)}$	Distortion of the ROI of frame f
$D_{ROI,SP}^{(f)}$	Distortion of the ROI of the spatially filtered frame f
$E\{\cdot\}$	$E\{\cdot\} = \frac{1}{N(f)} \sum_{(m,n)}$
$E_C\{\cdot\}$	$E_C\{\cdot\} = \frac{1}{N_C(f)} \sum_{(m,n) \in C}$
$E_p(m,n)$	The value of the parameterized ellipse used to indicate if pixel (m,n) contains skin color.
f	Frame index $f = 1, 2, \dots, F$.
$filt$	The filter $filt \in \{SP, TP, SPTP\}$ is applied to the sequence.
F_{rate}	Frame rate in frames per second (fps).
F	Total number of frames.
$G^{(i,j)}$	A gaussian centered at $(i,j) = (0,0)$ and with standard deviation σ .
$G^{(s)}$	The s :th gaussian filter in the spatial filter with standard deviation σ_s .
$H^{(s)}$	The interval of the quality map Q corresponding to filter $G^{(s)}$.
I_{odd}	Odd frames.
I_{even}	Even frames.
$I^{(f)}$	Frame with index f .
$I^{(f,(m,n))}$	Value of pixel (m,n) in frame f .
$\hat{I}^{(f,(m,n))}$	Value of pixel (m,n) in reconstructed frame f .
$I_{Bg}^{(f,(m,n))}$	Value of pixel (m,n) in the background of the frame f .
$I_{Bg,SP}^{(f,(m,n))}$	Value of pixel (m,n) in the background of the spatially filtered frame f .
$I_{Bg,TP}^{(f,(m,n))}$	Value of pixel (m,n) in the background of the temporally filtered frame f .
$I_{Filt}^{(f,(m,n))}$	Value of pixel (m,n) in the filtered frame f .
$\hat{I}_{Filt}^{(f,(m,n))}$	Value of pixel (m,n) in the filtered and reconstructed frame f .
$I_{even}^{(m,n)}$	The value of the pixel (m,n) in the even frame.

$\mathbf{I}_{odd,SP}^{(m,n)}$	The value of the pixel (m, n) in the spatially filtered odd frame.
$\mathbf{I}_{even}^{(p,q)}$	The block $B_{TP}(p, q)$ in the even frame.
$\mathbf{I}_{even,SPTP}^{(p,q)}$	The spatio-temporally filtered block $B_{TP}(p, q)$ of every even frame.
$\mathbf{I}_{odd}^{(p,q)}$	The block $B_{TP}(p, q)$ in the odd frame.
$\mathbf{I}_{odd,SP}$	Spatially filtered odd frames.
$\mathbf{I}_{odd,TP}$	Temporally filtered odd frames.
$\mathbf{I}_{odd,SP}^{(p,q)}$	Spatially filtered block $B_{TP}(p, q)$ in odd frame.
$\mathbf{I}_{TP}^{(p,q)}$	The temporally filtered block $B_{TP}(p, q)$.
$J \times J$	Size of the kernel of the gaussian G used to create the quality map Q from the binary detection map B_{DM} .
K	Control parameter in equation (3.8).
(k, l)	Index of makroblocks within a frame.
$L \times L$	Size of 2D Gaussian filter $G^{(s)}$.
(m, n)	Index of pixels within a frame, where $m = 1, 2, \dots, M$ and $n = 1, 2, \dots, N$.
$M \times N$	Size of one frame.
m_{MV}	The mean value of the lenght of the motion vector components.
$m_{C,MV}$	The mean value of the lenght of the motion vector components in region C .
m_{Diff}	The mean value of the difference between votes when the same pair is assessed twice for one test subject.
\bar{M}_{Ne}	The neighbor mean value map, where the mean value of block (u, v) is weighted by the average mean values of its neighboring
$\bar{M}_{Ne}^{(u,v)}$	The block $B_V(u, v)$ in the neighbor mean value map \bar{M}_{Ne} .
$\bar{M}_{Ne,SUM}$	The sum of the mean values of the neighboring blocks of block $B_V(u, v)$.
$\bar{M}_{J \times J}^{(m,n)}$	The mean value of using a filter kernel of size $J \times J$ centered at (m, n) .
$\bar{M}_{U \times U}^{(u,v)}$	The mean value of $B_V(u, v)$ using a filter kernel of size $U \times U$.
m_{VOTE}	The mean value of the votes in the subjective test.
$N_C^{(f)}$	Number of pixels within region C in frame f .
N_{Bg}	Number of pixels in the background of frame f .
$N_{Bg,Skip}$	Number background pixels which are excluded from the spatial filter by the variance measure in frame f .
N_{mv}	Number of motion vectors in the sequence.
$N_{C,mv}$	Number of motion vectors within the region C of the sequence.
$N_{mv}^{(f,(k,l))}$	Number of motion vectors within MB (k, l) of frame f .

$N_{C,mv}^{(f,(k,l))}$	Number of motion vectors within MB(k, l) of region C in frame f .
N_{ROI}	Number of pixels in the ROI.
N_{Tr}	Number of pixels in the transition area of the even frames.
Q	Quality map.
$Q^{(m,n)}$	The value of the quality map for pixel (m, n) .
Qp	Parameter controlling quantization step size in the encoder.
$Q^{(p,q)}$	Maximum value of $Q^{(m,n)}$, where $(m, n) \in B_{TP}(p, q)$.
Q_{var}	The quality map modified using a variance measure.
(p, q)	The index of the 8×8 blocks used in the temporal filter, where $p = \lceil m/8 \rceil$ and $q = \lceil n/8 \rceil$.
$P_{error,MSE}$	The prediction error in equation (3.1) with MSE as distortion measure.
$P_{skin}(c)$	The probability of observing the color c knowing that a skin pixel is observed.
$P(skin c)$	The probability of observing skin given a color value c .
$PSNR_{Avg}$	The mean peak signal to noise ratio of all frames in the sequence.
$PSNR_{Border,Avg}$	The mean peak signal to noise ratio of the border of all frames in the sequence.
$PSNR_{dB}$	Peak signal to noise ratio in decibel.
$PSNR_{ROI,Avg}$	The mean peak signal to noise ratio of the ROI of all frames in the sequence.
R_{target}	Target bit rate kbps in frames per second (fps).
R^f	Total number of bits in kb assigned to frame f .
$R_{Bg}^{(f)}$	Number of bits assigned to the background of frame f .
$R_{Bg,DCT}^{(f)}$	Number of bits assigned to the DCT components in the background of frame f .
$R_{Bg,SP,DCT}^{(f)}$	Number of bits assigned to the DCT components in the background of the spatially filtered frame f .
$R_{Bg,OH}^{(f)}$	Number of bits assigned to the overhead bits, which are always present, in the background of frame f .
$R_{Bg,MV}^{(f)}$	Number of bits assigned to the motion vectors in the background of frame f .
$R_{Bg,PErr}^{(f)}$	Number of bits assigned to the prediction error in the background of frame f .
$R_{Bg,SP}^{(f)}$	Number of bits assigned to the background of the spatially filtered frame f .
$R_{Bg,SP,MV}^{(f)}$	Number of bits assigned to the motion vectors in the background

	of the spatially filtered frame f .
$R_{Bg,SP,PErr}^{(f)}$	Number of bits assigned to the prediction error in the background of the spatially filtered frame f .
$R_{Bg,TP}^{(f)}$	Number of bits assigned to the background of the temporally filtered frame f .
$R_{Bg,TP,OH}^{(f)}$	Number of bits assigned to the overhead bits, which are always present, in the background of the temporally filtered frame f .
$R_{Bg,TP,MV}^{(f)}$	Number of bits assigned to the motion vectors in the background of the temporally filtered frame f .
$R_{Bg,TP,PErr}^{(f)}$	Number of bits assigned to the prediction error in the background of temporally filtered frame f .
$R_{ROI}^{(f)}$	Number of bits assigned to the ROI of frame f .
$R_{ROI,SP}^{(f)}$	Number of bits assigned to the ROI of the spatially filtered frame f .
s	Index of gaussian filters used in the spatial filter, where $s = 1, 2, \dots, S$.
S	The total number of gaussian filters used in the spatial filter.
σ	Standard deviation of Gaussian filter G .
σ_{Diff}	The standard deviation value of the difference between votes when the same pair is assessed twice for one test subject.
σ_{MV}^2	The variance of the motion vectors in the sequence.
$\sigma_{C,MV}^2$	The variance of the motion vectors of region C in the sequence.
σ_s	Standard deviation of the s :th Gaussian filter $G^{(s)}$ used in the spatial filter.
(u, v)	Index of the $U \times U$ block used for the variance map V , where $u = \lceil m/U \rceil$ and $v = \lceil n/U \rceil$.
T_m	The maximum allowed mean value difference between two neighboring blocks.
T_v	The lower threshold on variance effecting the spatial filter.
$U \times U$	Size of the block used for the variance map V and its mean value $\bar{M}_{U \times U}$.
V	Variance map.
$V^{(u,v)}$	The block $B_V(u, v)$ in the variance map V .
V_{erode}	Eroded variance map.
$V_{erode}^{(u,v)}$	The block $B_V(u, v)$ in the eroded variance map V_{erode} .
$Y^{f,(m,n)}$	The intensity component of the pixel (m, n) in frame f , assuming YCbCr colorspace.
$\hat{Y}^{f,(m,n)}$	The intensity component of the pixel (m, n) in reconstructed frame f , assuming YCbCr colorspace.

Chapter 1

Introduction

The main goal associated with video source coding is to reduce the amount of data used to describe the video sequence with as limited an effect on the quality as is possible. Region of interest (ROI) coding advances this concept by allowing higher quality within interesting regions of the video sequence without increasing the total amount of data. In low bit rate video transmission the necessary encoding causes reduced quality in regions of the video sequence with high detail and motion content. The perceived quality is experienced as particularly poor if the region contains information, which is important to the viewer.

ROI video coding methods increase the quality in regions of interest of the viewer at the expense of the quality in the background, compared to that when using ordinary encoding. In applications involving, for example, video conferencing, surveillance and transmission of sports, the interesting regions within the sequence can be identified. For example, in a video-conferencing sequence the face is of interest. Region of interest coding uses this information to apply different levels of compression to different parts of the sequence.



Figure 1.1: Video transmission system

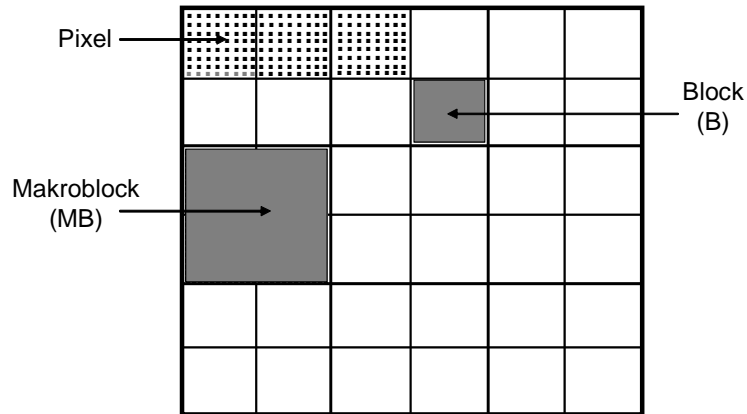


Figure 1.2: The partition of one frame in a video sequence into makroblocks (MB), blocks (B) and pixels.

1.1 Video coding

A digital video sequence consists of a limited number of images, which are called frames, and which are extracted at sufficiently small time interval to preserve the continuity in the sequence. Each frame is built up from small picture elements, pixels, which describe the color at that point in the frame using three different components. The result is a large amount of data which is necessary in order to completely describe the video sequence, which need encoding in order to enable transmission over channels with limited bandwidth. An example of a video transmission system can be found in fig. 1.1, where the sequence is compressed by the encoder before it is transmitted over the channel. In the decoding step the sequence is reconstructed. The reconstructed video sequence contains errors introduced both by the removed information in the compression and distortion in the channel.

1.1.1 Block-based hybrid coding

The existing video coding standards is based on block-based hybrid coding. The basic scheme partitions each frame within a video sequence into 16x16 pixel makroblocks (MB) and these MBs are further partitioned into blocks (B) of 8x8 pixels (See figure 1.2).

Each frame is either intra-coded using only information within the frame, or inter-coded, which indicates that information from other frames is also used. These

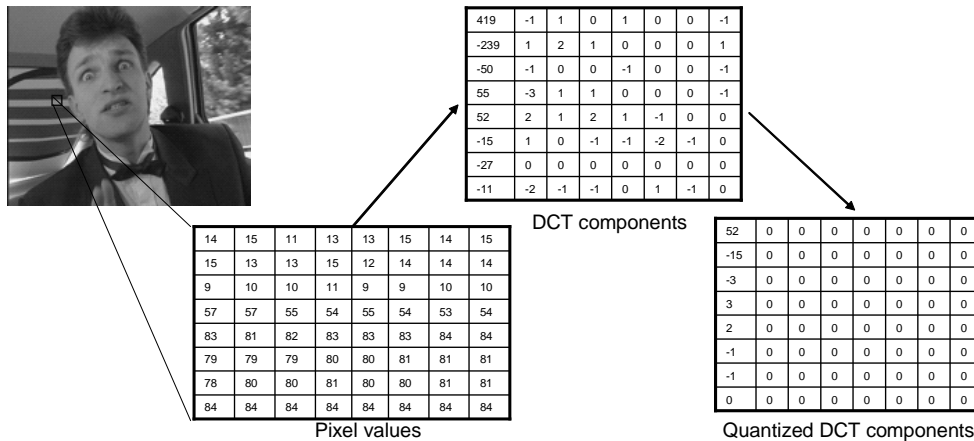


Figure 1.3: Discrete Cosine Transform (DCT) of a block within a frame followed by quantization of the resulting components.

two methods provide the schemes hybrid character.

In an intra-coded frame each B is subjected to transform coding using the Discrete Cosine Transform (DCT) (See fig 1.3). The DCT represents the information in each block using a mean value of the block and the deviation from this mean value using combinations of 63 different patterns. Video usually contains several one colored areas, which in this case would be represented as one value instead of 64 values. In the case of deviations from this mean value in general only a few of the 63 patterns are necessary to express this deviation. Therefore much less information needs to be transmitted with this representation than if the pixel values were transmitted directly. The resulting components after the DCT is quantized and encoded using a variable length code (VLC), before they are transmitted.

In the inter-coded frames the fact that only minor changes occur between two frames is used in the encoding. This change can be represented by motion vectors which contain the number of pixels by which a makroblock has moved compared to its best match in the previous frame (See fig 1.4). The error between the MB and its best match in the previous frame, the prediction error, is transformed using the DCT and thereafter quantized. The motion vectors and the quantized prediction error are encoded using VLC. The bit stream consisting of mainly VLC, is decoded by performing the reverse operations associated with the encoding of both the intra- and inter-coded frames in order to reconstruct the sequence at the receiver side.

The main goal associated with compression is to decrease the bit rate with as little

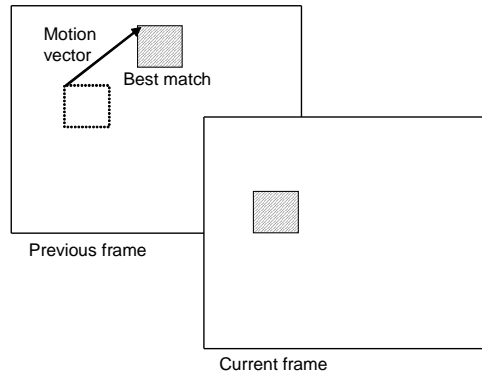


Figure 1.4: The best match from the previous frame is estimated and the motion vector to the position of the best match determined.

effect on the quality as is possible. The inter-coded P-frames results in less bits per frame than the intra-coded I-frames thus providing the motivation to use them in the encoding. On the other hand, I-frames enables access to the sequence at other times than at the first frame, which is assumed to be an I-frame. I-frames also prevents quality degradation, since errors propagates from P-frame to P-frame. In some cases B-frames are applied which predicts motion vectors from both the previous and the next frame.

1.1.2 Video coding standards

The Moving Pictures Reference Group MPEG [1], which is a working group of ISO/IEC and the International Telecommunications Union ITU [2] are responsible for producing the main standards available for video compression standards. These include MPEG-2 [3], H.261 [4] and H.263 [5], which are intended for different application with block-based hybrid coding as the common factor (See section 1.1.1). The standard MPEG-4 [6] includes additional features such as object-based coding (See section 1.3.3). The most recent standard is H.264 [7][8], also called MPEG-4 AVC, which has a higher compression efficiency than previous standards. The standards only describe the features supported by the encoder, define the syntax and semantics of the bit stream and the manner in which the transmitted bit stream should be parsed and decoded [9]. This enables some freedom when implementing the codec.

1.2 Perceptual quality

Video source coding in general aims to preserve quality, while reducing the bit rate. In most cases the quality is defined by the extent of the error introduced by the compression independent to its position in the video sequence. This is a simplification, which disregards the complexity of the human visual system (HVS). The perceptual quality is highly dependent on the information being transmitted at the location of the error. In regions containing details and particularly important semantic content, such as the face in video conferencing, the impact of an error is much greater than in the less important background. Methods for detecting quality are presented in section 2.4.

1.3 Region-of-interest video coding

At low bit rates video coding is performed in order to optimize the average quality under the constraints of a limited bandwidth. The results of this are that the quality in regions of the sequence containing high detail and motion content, such as a talking head, is reduced. This can be seen in fig 1.5.a, where the talking human is included in the ROI. When the most important information within the frame is found in these regions the perceived quality will be greatly affected. If these interesting regions can be detected ROI video coding can be applied, which increases the quality in the interesting region at the expense of the quality in the background as in fig 1.5.b. The result is an increase in perceived quality without increasing the bit rate.

The ROI video coding consists of two main steps. The ROI must firstly be detected, which requires previous knowledge of what a human would find interesting in the sequence. The perceived quality may even be reduced if the correct ROI is *not* detected. Secondly the video sequence is compressed using different amounts of encoding based on the detected ROI. This is achieved by bit allocation, which controls how many bits will be allocated to the different parts of the video sequence. More details concerning ROI video coding are presented in chapter 2.

1.3.1 Applications

Transmitting video conference material, as for example video conversations over the mobile phone, can be problematic for low bit rates. The encoding of regions of high motion and texture content gives an indiscriminately high number of artifacts. An example of this is the facial region in a videoconference sequence, where the

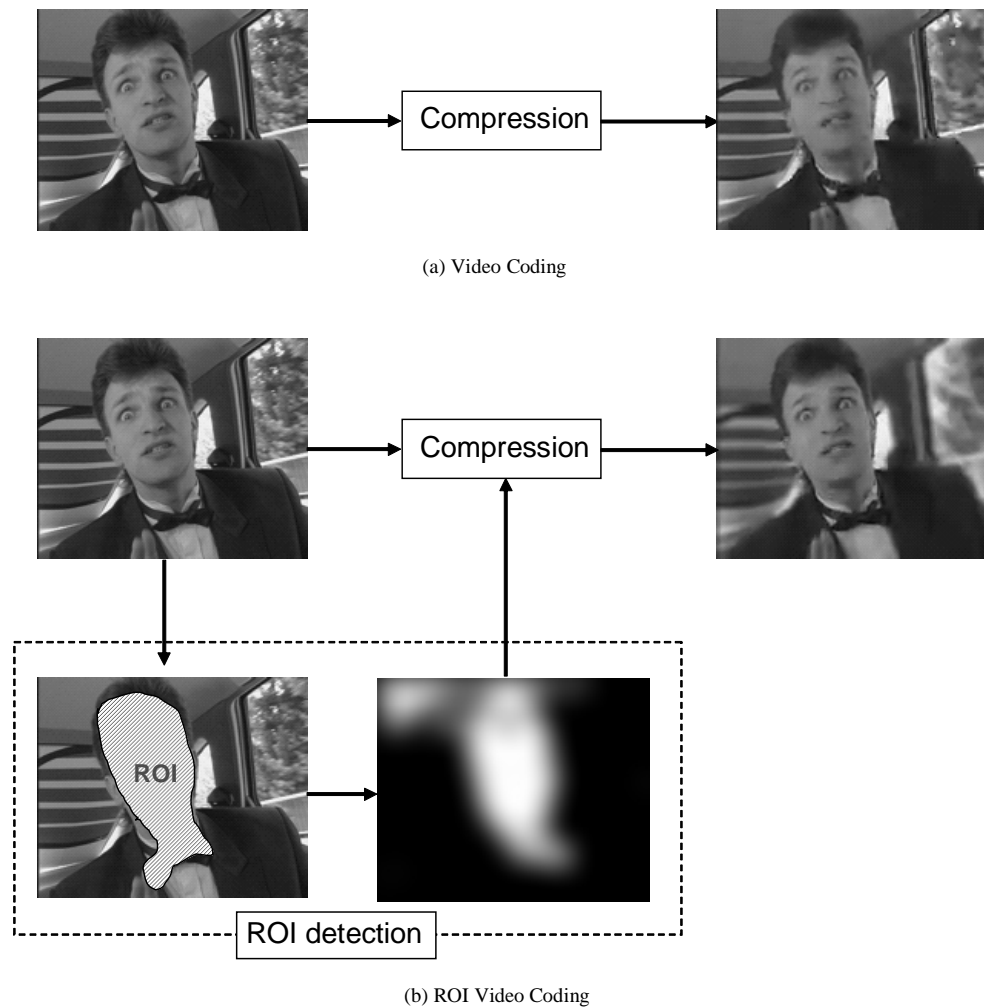


Figure 1.5: When ordinary video compression is applied, as can be seen for one frame in (a), all parts of the frame are compressed equally indifferent of content. However by using ROI video coding, as in (b), the perceived quality can be improved by compressing regions differently depending on content.

information communicated by the movement of the lips and facial expressions is lost if the artifacts are sufficiently large. In addition, reduced quality in the facial area can appear more disturbing for the viewer than reduced quality of the background. This is solved, in several approaches, by applying more compression to the background than in the facial region and is included in the articles by Eleftheriadis et al. [10] and Chen et al. [11].

In surveillance applications the regions of interest are often well defined, for example people or vehicles. The problem associated with analysing and transmitting surveillance video from several cameras in real time at low bit rates have been addressed in several publications. An approach which dealt with the means of detecting camouflaged enemies in the shape of people or military vehicles was introduced by Tankus et al. in [12], and which was suggested could be applied to ROI video coding in [13]. Other non-military examples of surveillance where ROI approaches have been applied include traffic surveillance in [14] and security applications in [15].

In [16] McCarthy et al. it was observed that in sequences from soccer games the quality of the player and ball proved to be the most important factor for the viewer. This indicates that by classifying the players and ball as ROI's improves the visual experience at low bit rates. Detection of ROI's in soccer sequences has also been attempted by Kang et al. in [17].

It is also possible to allow the user to define their own ROI and use the bit allocation methods presented in section 2.2 to control the quality in the video sequence.

1.3.2 Foveated coding

A related research area to ROI video coding is to use foveas instead of ROI's. In biology, the fovea is the part of the retina in the human eye which contains the greatest number of photoreceptors. Details in images can only be perceived if that part of the image is processed by the fovea. Thus only the point upon which the human gaze is currently fixed must be presented with good quality, enabling quality reduction based on the distance to this point. In the foveation approach foveas, which have a gradual quality reduction based on the distance to a point, are placed with their centers at the pixels where the human is predicted to gaze in each frame as in [18]. This approach demands that the exact location of a person's gaze is known, whereas in ROI coding it is only necessary to detect the region of the gaze.

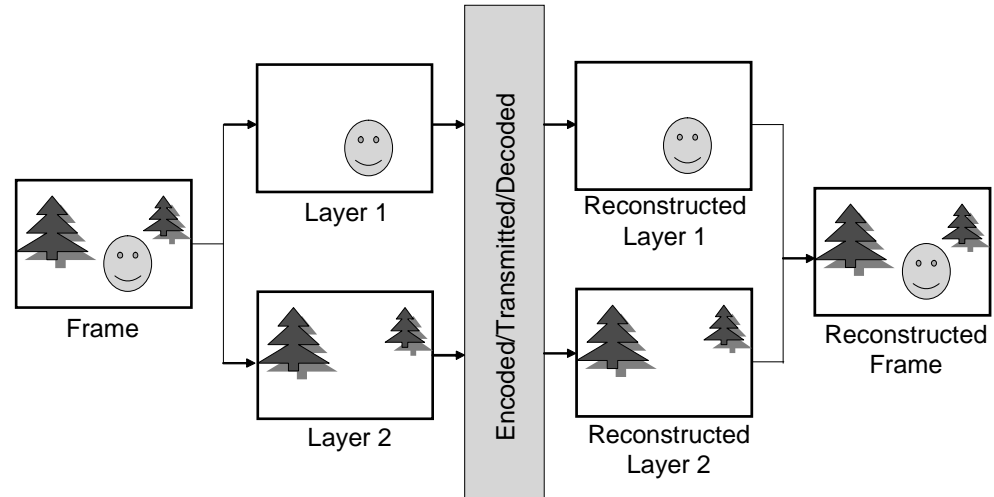


Figure 1.6: In object-based coding the different objects in a frame and the background are encoded in separate layers. The reconstructed layers are synthesized into one frame after the decoding of the layers.

1.3.3 Object-based coding

In the standard MPEG-4 [6] objects and background can be divided into a set of layers, which are compressed separately and then synthesised into one sequences at the receiver. In figure 1.6 the partition into background and object layers are described. Region-of-interest coding differs from object-based coding in that regions of different characteristics are extracted instead of specific objects. In addition in most region-of-interest cases the region and background are not separated into layers, but are instead transmitted as an ordinary video sequence.

1.4 Overall aim

The primary goal of this licenciate thesis is to increase the percieved quality of video sequences with a known ROI at low bit rates without introducing changes to the standard or encoder.

1.5 Scope

The percieved quality is improved by applying the ROI video coding concept and is limited to methods which decrease the information in the background enabling more

information to remain within the ROI without increasing the bit rate. It is assumed that the ROI has already been successfully detected. The scope of this thesis is limited to pre-processing method, since methods associated with altering the encoder means that the implementation must be redone each time the encoder is changed. Pre-processing methods in both the spatial and temporal domain are investigated.

Only sequences involving communications with humans are considered in the tests of the methods in order to limit the impact of false detections of the ROI. Thus only sequences where the use of the face detection algorithm was succesful were applied. However, in practice as long as the ROI detection is successful it can be assumed that the methods could be applied to any type of ROI. Only source coding is investigated and therefore no transmission errors are considered in the tests. The performance of the three filters are evaluted by a qualitative analysis of the effect the filters have on the encoding and on computational complexity. In addition quantitative tests using both objective measures are performed and subjective measures are used to verify the results.

1.6 Outline

This thesis is organized as follows: In chapter 2 an overview of the related works in ROI video coding field is presented. This is followed by three chapters 3, 4 and 5, where three filters are proposed and presented together with an analysis of their effect on the rate-distortion optimization of the codec, computational complexity and a presentation of the experimental results. These three filters are the spatial filter in chapter 3, the temporal filter in 4 and a combination of the previous two into a spatio-temporal filter in chapter 5. Thereafter a comparision of the three filters can be found in chapter 6 followed by conclusions in chapter 7.

1.7 Contributions

The contributions to this licenciate thesis consists of:

- Improvement of existing spatial filters to ensure that artifacts at the ROI border are avoided including a method concerning the means by which the computational complexity of the spatial filter can be reduced. (See chapter 3)
- Introducing a temporal filter which causes the encoder to skip the background in every second frame with bilinear interpolation to ensure that artifacts of the moving ROI border are reduced. (See chapter 4)

- A theoretical presentation of the effects of spatial and temporal filtering on the rate and distortion of the encoder considering block-based hybrid encoding and different standards. (See sections 3.1, 3.3 and 4.2).
- A combination of the spatial and temporal filters which increases the coding efficiency further and reduces the computational complexity. (See chapter 5)
- A comparison of the performance of the three filter types. (See chapter 6)

Chapter 2

ROI video coding

The idea of increasing quality within the ROI by decreasing quality in the background is called ROI video coding. This can be divided into two separate steps. Firstly the ROI is detected by predicting the type of content in a region that attracts the viewer's gaze and communicates the greatest amount of information. Based on these characteristics the position of the ROI is extracted (See section 2.1). In the second step the bit allocation is controlled in order to ensure that more bits are allocated to the ROI to increase the visual experience (See section 2.2).

2.1 ROI detection

The key to a successful ROI video coding is to correctly predict and detect the ROI, since a falsely detected ROI gives a lower perceptual quality than for ordinary video coding at low bit rates. This is achieved either by applying a generalized or an application-based approach. The generalized approach is based on visual attention models presented in section 2.1.1. In the application-based approaches the type of content present in an interesting region is predicted apriori for a particular application. These include video conferencing and videophone applications where faces are of interest (See section 2.1.2), surveillance of people and vehicles or in sports applications. Applications of ROI coding are further addressed in section 1.3.1.

2.1.1 Visual attention

Visual attention models determines the likelihood that a human viewer fixes his/her gaze on a particular position within a video sequence. This is generally based on models of the HVS. In [19] and [20] measures of color, orientation, direction of move-

ment and disparity are combined into a saliency map indicating the probability of this pixel drawing attention. Bollman et al. in [21] and Ho et al. in [22] extended these approaches by including motion detection. In addition Ho et al. in [22] also includes face detection. These approaches assume that the visual attention models can be generally applied however Ninassi et al. in [23] showed that the positions within an image where people are gazing are effected by the given task. This implies that the detection would be more accurate when the task is considered in detection, which is often the case in application-based approaches.

2.1.2 Face detection

In video communications human faces communicate a substantial part of the unspoken information using facial expressions and lip movements. Improving the quality of the human face by applying ROI video coding reduces the number of errors when interpreting the information and gives an experience of increased quality to the viewer. Face detection is a popular area including research on face recognition for surveillance and biometric authentication, coding using 3D-models and applications, where facial expression is extracted and analysed. In [24], Yang et al. present an overview of methods used for face detection including feature-based methods, template matching, eigenfaces, neural networks, support vector machines and more.

Most ROI video coding approaches apply different skin-color detection methods which are presented in 2.1.2. Skin-color detection is a fast method which is invariant to pose, orientation, difference in skin-color and occlusion. However skin-color can be combined with other feature detection methods to ensure robustness with respect to illumination changes and increased selectivity. Other face detection for ROI video coding approaches include extracting facial countours from edges in [25] and using a the unsupervised neural network described as a self-organizing map (SOM) as in [26].

Skin-color

An overview of pixel based skin detection techniques including different color spaces and classification methods can be found in [27]. A wide variety of color spaces have been used for skin-color detection including RGB, normalized RGB, HSI, HSV, HSL, TSL, YCbCr and others, where different color maps are suitable for different purposes. However, YCbCr is a common choice, because it separates luminance and chrominance and it is used in most video compression standards. YCbCr can be calculated from RGB using the simple transformation

$$\begin{aligned}
Y &= 0.299R + 0.587G + 0.114B \\
C_r &= R - Y \\
C_b &= B - Y
\end{aligned} \tag{2.1}$$

Several methods can be used to classify skin color based on a particular color space, including skin cluster boundaries, normalized lookup tables (LUT), Bayes classifiers, elliptic models, single Gaussians models and mixtures of Gaussians models.

The simplicity of describing the boundaries of skin clusters with boundary rules [11] [28] [29] makes this type of detection attractive. However, it is difficult to empirically find adequate color spaces and decision rules. As investigated in [30] compactly clustered skin-color in some color spaces enables the use of parametric models as single Gaussians and Gaussian mixture models for the approximation of the probability density functions (pdf) of skin-color. Parametric methods can be represented by a few parameters and have the ability to interpolate over incomplete training data. Single Gaussian models as in [30, 31] and the related elliptical model in [32] are very quick in training and classification but their strict shape limits performance. The Gaussian mixture model [33], on the other hand, improves detection accuracy at the cost of increased training and classification time.

According to Yang et al. in [24] and Vezhnevets et al. in [27] the dependence of color space by the parametric methods is avoided by applying non-parametric methods instead, at the cost of increased storage space and higher requirements on the training set. Non-parametric approaches include LUT in [34], which is a normalized histogram representing the probability of observing the color c knowing that a skin pixel is observed, $P_{skin}(c)$. A more appropriate skin-color detection measure would be to observe the probability of observing skin given a color value c , $P(skin | c)$. This probability can be determined by applying Bayes rule as in [35] and [36] and the Bayes classification is shown in [37] to give the highest accuracy in detection compared to other pixel based methods.

The robustness to illumination conditions can be improved by adapting the skin-color detection to illumination changes. In [36] Phung et al this is attempted by adaptively updating the Bayesian probabilities and in [38] Zhu et al. adaptive updates of a Gaussian mixture model. Another approach by Hsu et al. in [39] applies a nonlinear adaptable color space determined from a white reference.

Multiple features

A problem associated with skin-color detection is that skin-color-like background items are likely to be detected. The presence of a face can be further verified using

additional features. These features include eyes and face boundary in [39] and [29], mouth in [39], size of ROI and number of holes in the region in [31] and homogeneity measure in [36] and [40]

2.2 Bit allocation

Reallocation of bits from the background to the ROI can be achieved either by a pre-processing stage before the encoding or by controlling the parameters in the encoder. The video coding standards allow alterations of the encoder as long as the required features are included and the syntax of the bitstream is unaltered. However, each time the encoder is changed any modifications must be remade and adapted to the new encoder. This can be avoided by using pre-processing instead, but with loss of adaptability of background quality to variable bit rates of the channel.

2.2.1 Spatial bit allocation

The majority of research on bit allocation for ROI video coding apply spatial methods and in particular those controlling parameters within the codec. The quantization step sizes controls the quantization accuracy concerning DCT components and the prediction error. In [41] and [11] two step sizes are used for each frame. A small step-size is used for the ROI and a larger stepsize is used for the background. This results in a more finely tuned quantization and therefore improved quality of the ROI. However, the difference between background and ROI quality appears abrupt if there is a large difference in quantization. Solutions include adaptation of the quantization step sizes of the background to the distance of the ROI border [42, 43], applying three quantization levels in [25] or deciding the step size based on a sensitivity function [44]. Approaches involving other types of encoder parameters are addressed in [45] and [46], where the number of non-zero DCT components are used to control bit allocation.

Controlling quantization parameters allows direct integration with the rate-distortion function within the codec, but can introduce "blockiness" due to coarse quantization. Pre-processing, whose resulting error is generally less disturbing, avoids codec dependencies. Methods based on low-pass filtering (blurring) of non-ROI are addressed in [11] and [18]. Low pass filtering reduces the amount of information which gives less non-zero DCT components and a reduction in prediction error due to the absence of high frequencies. The rate-distortion optimization of the encoder then re-allocates bits to the ROI, which still contains high frequencies. In [11] Chen et al. low pass filtering of the background regions are applied using one filter for the

complete frame. This gives a distinct boundary separating the ROI and background which leads to disturbing artifacts. In the foveation coding approach by Itti in [18] a gradual transition of quality from the fixation point, where the human is predicted to gaze, to the background is achieved using a Gaussian Pyramid. Foveated Coding is addressed in section 1.3.2.

2.2.2 Temporal bit allocation

The number of bits necessary to encode the background is reduced if a lower frame rate (fps) is used, at the expense of a decrease in the quality of the background. The related object-based video coding addressed in section 1.3.3 extracts and encodes objects and background in separate layers, which are synthesized into one video sequence at the decoder. The layers can be encoded using different frame rates enabling reduced frame rates for the background as in [47]. This is supported by the MPEG-4 standard [6]. A similar approach is suggested in [14] by Meessen et al where the ROI and background are encoded and transmitted as two separate sequences requiring adaptation at the receiver side.

Compatibility with the standard remains if the temporal bit allocation is performed without affecting the syntax of the bit stream [48, 25, 49]. In [48] all blocks not used in the encoding of the ROI are skipped in the P-frames and their DCT coefficients are deleted in the I-frames. Lee et al. in [25] reduces the transmitted information by skipping background makro blocks in every second frame unless the global motion in the frame exceeds a threshold. A similar approach is presented by Wang et al. in [49], where the background blocks are skipped based on the content in the ROI and background.

Adiono et al. presents a pre-processing approach in [45], that applies a temporal average filter in order to average out differences between the background of two frames.

2.2.3 Combinations of spatial and temporal bit allocation

The spatial methods reduces the background information transmitted in DCT components or the motion prediction error. The temporal filters, on the other hand, mainly reduces the bits assigned to the background motion vectors, except in the case of the average filter in [45] which affects the prediction error instead. Thus combinations of spatial and temporal approaches increases the reallocation of bits from the background to the ROI, since the spatial filters and the temporal filters reduce the background bits in different parts of the encoding. In [25] Lee et al. combine a spatial method controlling quantization step sizes with the skipping of background

blocks for every second frame under limited global frame motion. Spatial methods controlling the number of DCT components are combined with a similar temporal method as [25], but adapted to ROI and background content in [49], and combined with an temporal average filter in [45].

2.3 Other applications of ROI

The region of interest concept has applications other than those related directly to video source coding. One method of coping with errors introduced by the channel in the transmission is data partitioning, which introduces unequal error protection to the video sequence. In [50] Hannuksela et al suggests partitioning the data into ROI and background to ensure more error protection for the ROI. The ROI related approach foveation in section 1.3.2 has also been suggested as a tool for controlling data partitioning in [51]. Reduction of the effect of channels errors has also been addressed by Hannuksela et al. in [52], where the prediction of ROI blocks is restricted to other ROI blocks in the current frame or in neighboring frames.

In addition the use of ROI in transcoding applications has also been suggested. In [53], Lin et al. apply ROI in order to reduce the frame rate of non-active users in multi-point video conferencing. Dogan et al. in [54] suggests the use of ROI when downscaling high quality video to enable transmission in a heterogenous network.

2.4 Quality measures

Quality assessment of video is usually performed with fast and repeatable objective measures, which uses predefined algorithms. However, the algorithm fails to include all aspects of the HVS and therefore does not give a precise measure of perceived quality. Subjective tests, with human subjects, gives a good representation of the HVS but is time-consuming and has high requirements with regards to the test setup in order to ensure analysable tests.

2.4.1 Objective quality

Video coding quality is in the majority of cases measured using the mean square error (MSE) based peak-signal-to-noise ratio (PSNR) [55]

$$PSNR_{dB} = 10 \log_{10} \frac{255^2}{MSE}$$

$$MSE = \frac{1}{N} \sum_{i=1}^N (x_i - y_i)^2,$$

where N is the number of pixels within the video signal, 255 is the maximum pixel value and x_i and y_i represents pixel i in the original and distorted video sequence, respectively. This measure only considers errors at the pixel level while completely disregarding the position of the pixel or the context of the region. ROI video coding aims to improve perceived quality by increasing the quality in the ROI at the expense of that in the background and therefore the PSNR of the ROI is often analysed by itself. This gives an indication of how much the quality increases within the ROI, but no indication of the effect of decreased background quality to the perceived quality. Attempts have been made to include more HVS characteristics in objective measures, which are summarized by Wang et al. in [56]. Several different approaches have been considered such as applying foveation and visual attention [57], contrast sensitivity [58] and structural distortion measurement instead of error measurement [59] and several others.

2.4.2 Subjective quality

Subjective tests using human subjects enables tests measuring the actual perceived quality. Successful extraction of analysable results from subjective tests depends on several factors including the test time, test order and the instructions received. In addition even if one test person experiences good quality this is not necessarily the case for the next test person. The standard ITU-R BT.500-10 [60] presents several different test methodologies concerning, for example, quality assessment of single video sequences and pairwise comparison of video sequences. The definitions include setup parameters and description of tests and scales used by the test subject to describe the perceived quality. The ITU-R quality scale in table 2.1 is used for evaluating single sequences, while the ITU-R comparison scale in table 2.2 can be applied to the comparison of two video sequences. Smaller subjective tests can be used to verify the results of objective measures.

Applying the standard enables comparison with other research results.

Vote	Quality
5	Excellent
4	Good
3	Fair
2	Poor
1	Bad

Table 2.1: The ITU-R Quality Scale used to evaluate the subjective quality of one video sequence.

Vote	Quality
-3	Much worse
-2	Worse
-1	Slightly worse
0	The same
1	Slightly better
2	Better
3	Much better

Table 2.2: The ITU-R Comparison scale used to evaluate the quality of one video sequence compared to another.

Chapter 3

Spatial filtering

In this chapter a method, which seeks to improve the codec independent bit reallocation in the spatial domain is presented. (See section 2.2.1.) The proposed spatial (SP) filter presented in figure 3.1 improves the ROI approach in [11] by allowing a gradual degradation in quality from ROI to the background in order to reduce border effects. Border effects appears when the SP filter causes large differences between neighboring ROI and background pixels at the ROI border. This is solved by extending the idea of using different Gaussian filters depending on the distance to the border of a region instead of the distance to a point of fixation as in the foveated coding approach presented by Itti in [18]. ROI video coding does not detect the exact position of the human gaze and therefore equal quality within the ROI is necessary. The basics involved in video compression are presented in section 3.1 and in particular the rate and distortion calculations used to create and analyse the algorithms. More details concerning the SP filter can be found in section 3.2.

The performance of this filter is evaluated both theoretically and using qualitative tests. A theoretical analysis of how SP filtered video affects the rate distortion optimization in the codec compared to encoding the original video sequence is presented in section 3.3. In addition a theoretical analysis of the computational complexity of the filter can be found in section 3.4. The results for the qualitative test measuring the performance using bit rate, PSNR and subjective tests are presented in section 3.6.

3.1 Block-based hybrid coding of a video sequence

In this section the description of block-based hybrid video coding in section 1.1.1 is explained in greater detail. In particular the rate and the distortion of the sequence,

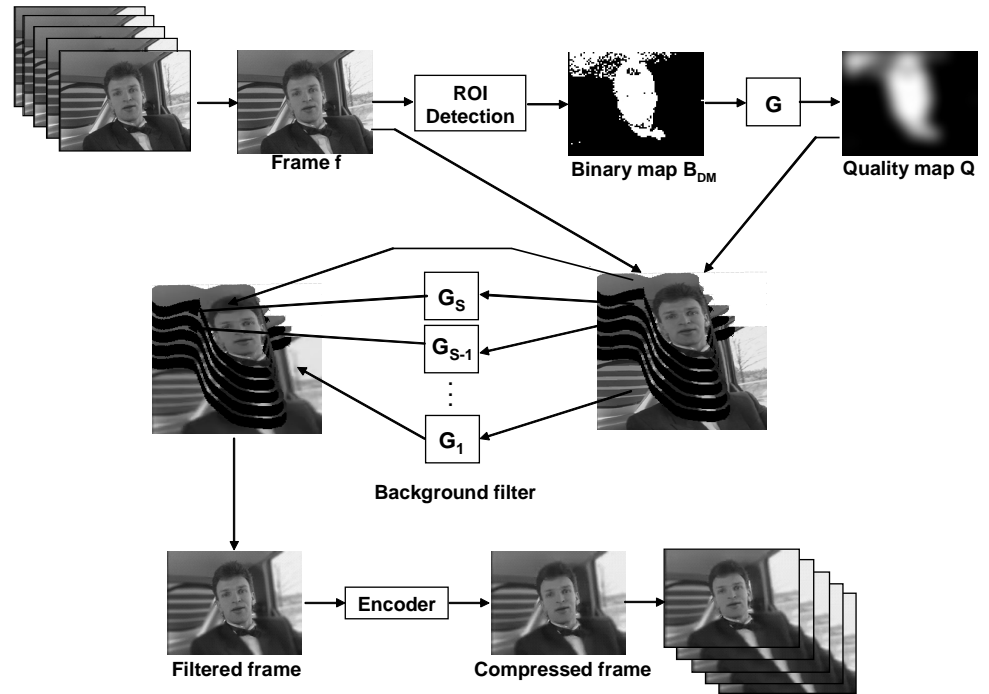


Figure 3.1: The spatial filter including the ROI detection, creation of the quality map and filtering of the background with S filters of variable variance σ_s^2 .

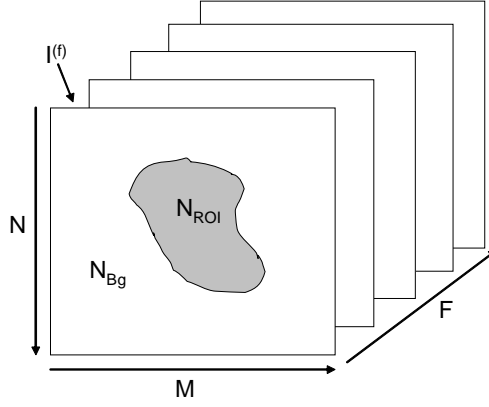


Figure 3.2: The video sequence is defined by the number of frames F and each frame is further defined by the number of pixels in each column M and each row N . In addition for ROI video coding each pixel either belongs to the ROI or the background and the number of pixels in the ROI and the background of frame $I^{(f)}$ are given by N_{ROI} and N_{Bg} , respectively.

since these are used to control the performance of the encoder.

In this thesis a video sequence is defined as F frames $I^{(f)}$ with index $f = 1, 2, \dots, F$ (See figure 3.2). Each frame contains M columns and N rows of pixels, where the index of a pixel is (m, n) for $m = 1, 2, \dots, M$ and $n = 1, 2, \dots, N$. Each frame is further divided into the ROI and the background, where the number of pixels assigned to the ROI and the background are given by N_{ROI} and N_{Bg} , respectively. In most color video coding systems the original video is described using the YC_bC_r color space described by equation (2.1). Therefore each pixel (m, n) in the original frame and in the compressed and reconstructed frame is represented by $\mathbf{I}^{(f, (m, n))}$ and $\hat{\mathbf{I}}^{(f, (m, n))}$, respectively and contains three color components $Y^{(f, (m, n))}$, $C_b^{(f, (m, n))}$ and $C_r^{(f, (m, n))}$. The commonly used video coding standards (See section 1.1.2) apply block-based hybrid coding as described in figure 3.3, where each block is either intra-encoded or inter-coded. The intra-coded blocks are first subjected to DCT as described in section 1.1.1, which represents the block in frequencies instead of pixels. This removes redundancy within the blocks enabling less information to be transmitted. The resulting DCT components are quantized (QUANT) and encoded using variable length codes (VLC). Inter-coded frames are encoded using prediction from the previous block. This is achieved by estimating the motion of that block (ME) compared to that of the previous frame, which is found in the previous frame memory (PFM). The motion estimation is performed for each MB by finding the \bar{d}_{MV} for which the prediction error

$$E\{|\mathbf{I}^{(f, (m, n))} - \mathbf{I}^{(f-1, (m, n) + \bar{d}_{MV})}|^2\} \quad (3.1)$$

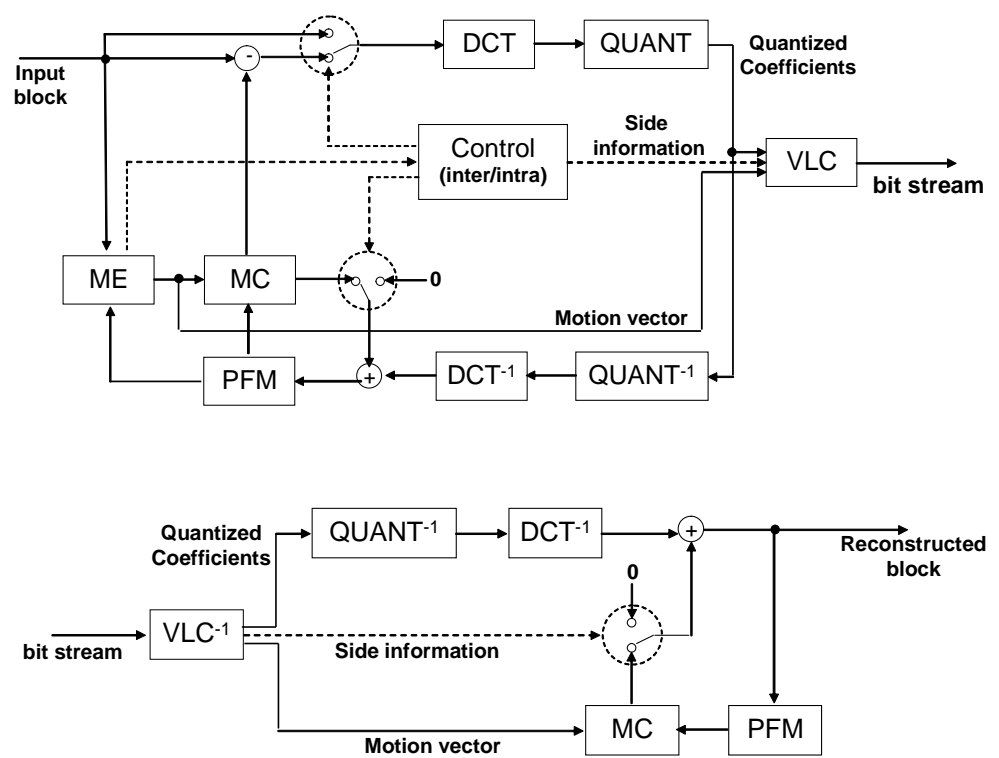


Figure 3.3: The encoding (top) and decoding (bottom) in a basic block-based hybrid coding system.

is minimized for that MB and $E\{\cdot\} = \frac{1}{N_{block}} \sum_{block}(\cdot)$. The sum of absolute differences (SAD)

$$\sum_{(m,n) \in block} |\mathbf{I}^{(f,(m,n))} - \mathbf{I}^{(f-1,(m,n)+\bar{d}_{MV})}|$$

is sometimes applied instead of MSE, since it has a lower computational complexity. However in this analysis it is assumed that MSE is applied, since the choice of MSE or SAD would not affect the result of the analysis. The resulting motion vector \bar{d}_{MV} is encoded using VLC. In the motion compensation (MC) part, the error between the current block and its best match in the previous frame is calculated. The position of the best match in the previous frame is given by translating the current position using \bar{d}_{MV} . This error is called the prediction error in this thesis. The resulting prediction error is encoded as an intra-coded block using DCT, quantization and VLC.

The encoding is controlled by a set of parameters, such as for example the quantization step size and the search range for the ME. These parameters can either be set manually, giving variable bit rates or determined using rate-distortion optimization. In the case of rate-distortion optimization the parameters are determined by minimizing the distortion D , while maintaining a target bit rate R_{target} . This is achieved by optimization of the rate distortion function [9]

$$\begin{aligned} \min \quad & D^{(f)} \\ \text{subject to} \quad & R^{(f)} \leq \frac{R_{target}}{F_{rate}} \end{aligned} \quad (3.2)$$

$$D^f = E\{|\mathbf{I}^{(f,(m,n))} - \hat{\mathbf{I}}^{(f,(m,n))}|^2\}$$

for the distortion D introduced by the codec as can be seen in figure 3.4 assuming MSE as distortion measure and a given target bit rate R_{target} [kbps] and a given frame rate F_{rate} [fps]. In this thesis the concept of rate is extended to the number of bits required for each part, such as MB and frame, which is related to the bit rate in bits per second by the frame rate and the number of MBs per frame. The rate $R^{(f)}$ in kb per frame f is given by

$$R^{(f)} = R_{ROI}^{(f)} + R_{Bg}^{(f)}$$

where $R_{ROI}^{(f)}$ and $R_{Bg}^{(f)}$ are the number of bits used to encode the ROI and the background of frame f , respectively. The total distortion of the video sequence by the compression assuming MSE as distortion measure is

$$D^{(f)} = \frac{(N_{ROI}^{(f)} D_{ROI}^{(f)} + N_{Bg}^{(f)} D_{Bg}^{(f)})}{MN}$$

$$D_C^{(f)} = E_C\{|\mathbf{I}^{(f,(m,n))} - \hat{\mathbf{I}}^{(f,(m,n))}|^2\}$$

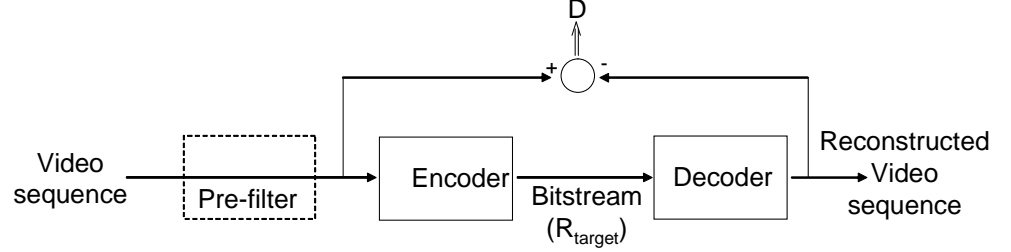


Figure 3.4: The distortion D considered in this thesis is only measured over the encoder/decoder pair and therefore when prefiltering is applied the filtered data is compared to the reconstructed data.

$$E_C\{\cdot\} = \frac{1}{N_C^{(f)}} \sum_{(m,n) \in C} (\cdot)$$

where N_c is the number of pixels within region $C \in \{ROI, Bg\}$ of frame f . The same distortion measure is applied to filtered signals where the distortion of region C in frame f becomes

$$D_{C,Filt}^{(f)} = E_C\{|\mathbf{I}_{Filt}^{(f,(m,n))} - \hat{\mathbf{I}}_{Filt}^{(f,(m,n))}|^2\} \quad (3.3)$$

where $filt \in \{SP, TP, SPTP\}$. The rate of each frame can be further divided into which parts of the coding they originate from. In an intra-coded frame the number of bits in the background can be divided into

$$R_{Bg}^{(f)} = R_{Bg,OH}^{(f)} + R_{Bg,DCT}^{(f)} \quad (3.4)$$

where $R_{Bg,OH}^{(f)}$ gives the number of overhead bits always present and $R_{Bg,DCT}^{(f)}$ is the number of bits assigned to the DCT coefficients in the background of frame f . Then for an inter-coded frame

$$R_{Bg}^{(f)} = R_{Bg,OH}^{(f)} + R_{Bg,MV}^{(f)} + R_{Bg,PErr}^{(f)}$$

where $R_{Bg,MV}^{(f)}$ gives the bits allocated to motion vectors and $R_{Bg,PErr}^{(f)}$ the bits used to encode the prediction error. In both intra and inter-coded blocks the $R_{Bg,OH}^{(f)}$ can be assumed to remain approximately the same even when the block is skipped in the inter-coded case.

3.2 The SP filter algorithm

The SP filter presented in figure 3.1 consists of three main parts. Firstly the detection of the ROI is described in section 3.2.1 and the means by which this information is used to calculate a quality map is described in section 3.2.2 The quality map contains

information concerning the position of the ROI and the distance to the ROI. This is used in order to control a set of Gaussian filters with different variances to create a gradual quality degeneration from the ROI to the background, which is addressed in section 3.2.3

3.2.1 ROI detection

The first step in the ROI coding determines the position of the ROI. Various detection methods can be used and the choice depends on the characteristics of the content of the ROI (See section 2.1). In this thesis we have chosen to focus on video-conferencing sequences. Thus the ROI is assumed be the face region, since it contains the most visual information of interest in a conversation.

The parametric model in [32], which is briefly described in appendix A, and an experimentally determined threshold is used in this paper to provide a binary detection map B_{DM} . This parametric model is used because of its simplicity, but it is highly dependent on lighting conditions and results in many false positives. The use of more accurate detection methods is recommended. In addition the threshold is determined experimentally for each sequence and thus a better decision method is necessary to make the detection general for all faces. The pre-filtering methods presented in this paper can be applied to any type of ROI and any number of ROIs under the assumption that the ROI detection gives a correct result. However the size of the ROI will affect how much data can be reallocated from the background to the ROI.

3.2.2 Quality map

The position of the ROI is used to determine where the filtering is performed and can be directly extracted from the binary detection map B_{DM} . A quality map Q indicating the distance to the ROI border is used to determine what filter to use. The distance is determined by filtering B_{DM} with a 2D Gaussian filter kernel of size $J \times J$ given by

$$G^{(i,j)} = \frac{1}{2\pi\sigma^2} e^{-(i^2+j^2)/2\sigma^2} \quad (3.5)$$

where σ^2 is the variance of the filter. The resulting quality map $Q^{(m,n)}$ contains continuous values between 0 and 1 indicating the importance of sustaining the quality at that location. The ROI and background are now redefined as

$$(m,n) \in ROI \text{ if } Q^{(m,n)} \geq A_{ROI}$$

$$Bg = \neg ROI$$

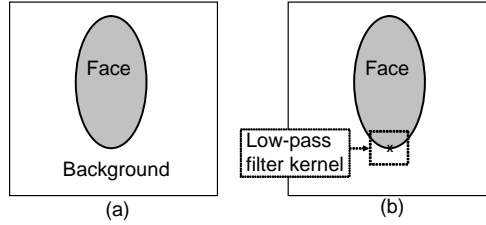


Figure 3.5: (a) The face is assumed to have an approximately oval shape. (b) More than 1/3 of the low pass filter kernel covers face pixels, when calculating the value for the border pixel in the thinnest part of the face.

The distance to the ROI border is indicated by $Q^{(m,n)}$. The minimum value of Q belonging to the ROI, A_{ROI} , is determined by assuming that faces are approximately oval in shape (See fig. 3.5.a) and that the filter kernel is less than half the size of the face. Faces less than twice the size of the filter kernel are sufficiently small to have unclear features. Low pass filtering of the binary detection map B_{DM} gives the lowest value of $Q^{(m,n)}$ for $(m,n) \in ROI$ at the ROI border and in particular where the face is the thinnest. It can then be assumed that at least a third of the pixels processed by the filter kernel for one such border pixel have the value 1 (See fig. 3.5.b). Thus $Q^{(m,n)} < A_{ROI} = 1/3$ ensures that the complete face is included within the ROI. This constant can also be applied to other types of ROIs with shapes similar to that of the face, however it might not be successful at detecting thin or small ROIs. The procedure for the quality map gives a larger ROI than in the B_{DM} and further removes high frequency components. Holes within the ROI are filled in and small ROI's are removed so no extra morphological filters are necessary. An alternative quality map may be utilized in order to reduce computational complexity. It is described in section 3.4.1.

3.2.3 SP filter

The background of each frame in the video sequence is filtered using a set of Gaussian filters with standard deviations which depend on the distance to the ROI. The distance is extracted from the quality map Q . The information in Q is used in the SP filtering by applying a set of thresholds partitioning the background values $[0, A_{ROI})$ into S intervals

$$H^{(s)} = \left[\frac{(s-1)A_{ROI}}{S}, \frac{sA_{ROI}}{S} \right) \quad (3.6)$$

for $s = 1, 2, \dots, S$, where S is the number of low-pass filters used. Each partition corresponds to one Gaussian filter $G^{(s)}$, which can be described as $G^{(x,y)}$ in eq. 3.5

with a variance σ_s^2 instead of σ^2 and size LxL . The variance σ_s^2 is used to control the impact of the different filters. Thus each pixel where $Q^{(m,n)} < 1/3$ and $Q^{(m,n)} \in H^{(s)}$ is low-pass filtered using filter $G^{(s)}$. The variances of the filters must have the property $\sigma_1^2 > \sigma_2^2 > \dots \sigma_{S-1}^2 > \sigma_S^2 > \sigma_{S+1}^2 = 0$ to ensure a gradual degradation in quality from the ROI to the interval $H^{(1)}$ filter with $G^{(1)}$. This property is fulfilled if the standard deviations of the filter are given values on a strictly decreasing curve such as the linear curve between σ_1 and $\sigma_{S+1} = \sigma_{ROI} = 0$ with respect to s . The standard deviation of the s :th filter σ_s is therefore given by

$$\sigma_s = \frac{\sigma_1(S + 1 - s)}{S} \quad (3.7)$$

The largest standard deviation σ_1 is used to control the degradation of quality in the background. In equation (3.7) the standard deviation of the ROI $\sigma_{S+1} = 0$ is used as an endpoint instead of σ_S in equation (3.8), which was used in a previous approach by the author. This ensures that the smooth quality degradation from $G^{(s)}$ to $G^{(s-1)}$ also apply from the ROI to those pixels filtered with $G^{(S)}$. The previously used alternative in equation (3.8) is also linear but in this case all of the standard deviations are altered by changing the control parameter K .

$$\sigma_s = K \left(\frac{(\sigma_1 - \sigma_S)(s - S)}{1 - S} + \sigma_S \right) \quad (3.8)$$

However this alternative is not used, since a badly chosen control parameter K in equation (3.8) could result in too large a value of σ_S thus causing an abrupt change in quality at the ROI border.

3.3 Rate-Distortion of SP filtered video

In this section the encoding of SP filtered video sequences will be addressed and compared to the encoding of the original video sequences. The analysis is based on the definitions in section 3.1. In section 3.3.1 we analyse the effect on rate and distortion by encoding filtered data instead of original data. A similar analysis of the inter-coded frames can be found in section 3.3.2.

3.3.1 Intra-coded frames

The SP filter removes all frequency components above the cut-off frequency. This implies that the number of non-zero DCT coefficients requiring to be quantized and encoded either decreases or remains the same in the background blocks. If we assume that the same quantization step sizes are used for both the original frame and

the filtered frame then $R_{Bg,SP,DCT}^{(f)} \leq R_{Bg,DCT}^{(f)}$, thus means that from equation (3.4)

$$R_{Bg,SP}^{(f)} \leq R_{Bg}^{(f)}$$

where $R_{Bg,SP}^{(f)}$ is the total number of bits allocated to the background of frame f in the spatially filtered sequence. Equality only applies when all frequencies in the background are below the cut-off frequency of the spatial filter. Thus when frequencies above the cut of frequency exist the bit rate is reduced.

The removal of high frequency components in the background means that

$$\begin{aligned} D_{Bg}^{(f)} &\geq D_{Bg,SP}^{(f)} \\ D_{ROI}^{(f)} &= D_{ROI,SP}^{(f)} \end{aligned} \quad (3.9)$$

for all frames f , where $D_{Bg,SP}^{(f)}$ is defined by equation (3.3). The distortion of the ROI is unchanged by the low pass filter while the distortion in the background is reduced. Thus the minimization in equation (3.2) implies that $R_{ROI}^{(f)} \leq R_{ROI,SP}^{(f)}$ and $R_{Bg}^{(f)} \geq R_{Bg,SP}^{(f)}$, since the overall distortion is reduced by assigning more bits to the ROI of the filtered sequences compared to those of the original sequence. Therefore the codec will automatically re-allocate bits from the background to the ROI improving the quality of the ROI, when applying the SP filter.

3.3.2 Inter-coded frames

Low pass filtering removes details, i.e. high frequencies, which means that for each MB the prediction error is the same or reduced compared to the prediction error of the original frame. An example of this is found in figure 3.6. Thus

$$E\{|\mathbf{I}_{Bg,SP}^{(f,(m,n))} - \mathbf{I}_{Bg,SP}^{(f-1,(m,n)+\bar{d}_{MV})}|^2\} \leq E\{|\mathbf{I}_{Bg}^{(f,(m,n))} - \mathbf{I}_{Bg}^{(f-1,(m,n)+\bar{d}_{MV})}|^2\} \quad (3.10)$$

where $\mathbf{I}_{Bg}^{(f,(m,n))}$ contains the value of the frame f when $(m,n) \in Bg$. Equality occurs when all frequencies are below the cut-off frequency of the spatial filter for both MBs or if the frequency components exceeding this value are identical in the two MBs. Otherwise by removing the detail in both blocks the difference between them decreases. The decrease in prediction error implies that the number of bits allocated to the prediction error of the background in the spatially filtered frame is reduced,

$$R_{Bg,SP,PErr}^{(f)} \leq R_{Bg,PErr}^{(f)}$$

for a fixed set of coding parameters, such as quantization step sizes and the search range for the ME.

The choice of motion vectors based on equation (3.1) is also effected by the spatial filter. The prediction of each MB is optimized by minimizing the sum of bits

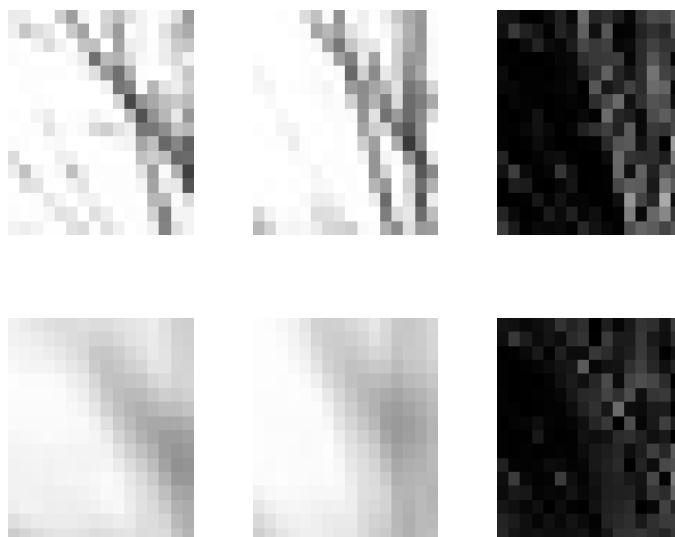


Figure 3.6: The prediction error of the MBs at the same position in two adjacent frames (top, left and (top, middle), $P_{error, MSE} \approx 2433$, is large because of the translation of the line from one MB to the other as can be seen in the difference of the two MBs (top, right). However if the two MBs are SP filtered (bottom, left and (bottom, middle), the difference between the MBs (bottom, right) is reduced significantly and the prediction error becomes $P_{error, MSE} \approx 202$.

allocated to encode both the motion vector and the resulting prediction error. Thus the choice of motion vectors might result in a reduction of bits when the prediction error is reduced. This occurs when the reduced cost of encoding this prediction error compared to the prediction error of the same MB in the original sequence is sufficient for the decision to be made to use the corresponding motion vector. That is the case in figure 3.6. The two blocks in the figure are in the same position in two adjacent frames. In the original frame it is apparent that the main detail in the shape of a line has moved. Due to the resulting high prediction error it is cheaper to encode the prediction error and the corresponding motion vector for the translation of that line than the prediction error without a motion vector. However, in the filtered case the prediction error has become sufficiently small for it to be encoded directly.

In addition, the number of motion vectors is related to the target bit rate, since motion vectors have a high cost in bits compared to other encoding components. A choice involving only the encoding of the prediction error (The motion vector is assumed to be $\vec{d}_{MV} = (0, 0)$.) could give an increase in the distortion that is sufficiently small such that it discourages coding both the motion vector and the prediction error if there is a shortage of bits. However, this motion vector can be used to reduce the overall distortion when more bits are made available by the SP filter.

Minimizing equation (3.2) indicates in a manner similar to that as for intra-coded frames that $R_{Bg}^{(f)} \leq R_{Bg,SP}^{(f)}$, since the prediction error of the spatially filtered background is smaller than for the original sequence. If the motion vector length is altered the sum of the bits allocated to the prediction error and motion vectors of that MB then become lower or the same as in the original sequence. Thus bits will be allocated to the ROI, where the prediction error is large and therefore can be improved.

3.4 Computational complexity

The computational complexity of the different parts of the SP filter is presented in this sections. In addition, versions of the algorithms, which aim at reduction in the computational complexity, are suggested. The computational complexity is defined as the number of operations, where an operation consists of addition, subtraction, multiplication or division. Absolute values are disregarded since they cost substantially less to perform. The computational complexity of the ROI detection depends on the detection method used (See section 2.1). However the skin-color detection in section 2.1.2 can be performed by Bayesian classifiers with two predetermined histograms [36] resulting in only one multiplication per pixel in the decision.

3.4.1 Quality map

Low-pass filtering with a Gaussian filter is necessary to create the quality map from a previously determined binary detection map B_{DM} . The size of the filter kernel, $J \times J$, must be sufficiently large to include all the pixels affecting the filtering when using this kernel. The filter has a computational complexity of $2J^2$ operations per pixel, but the separability property of the two dimensional Gaussian filter reduces the computational complexity to $4J$ operations per pixel by using two separate one-dimensional Gaussian filters. This results in a computational complexity of $4JMN$ per frame. If the mean value filter

$$\bar{M}_{J \times J}^{(m,n)} = \frac{1}{J^2} \sum_{a=m-\lfloor J/2 \rfloor}^{m+\lfloor J/2 \rfloor} \sum_{b=n-\lfloor J/2 \rfloor}^{n+\lfloor J/2 \rfloor} B_{DM}(a,b), \quad (3.11)$$

is applied instead of the Gaussian filter, the number of operations is reduced by 50%, with the effect that the distance measure now becomes linear. This linear distortion measure does not provide a rapid descent close to the ROI as does the Gaussian filter and therefore on average a slightly higher standard deviation will be used for the transition region.

3.4.2 Spatial filter

The S Gaussian filters used in the filtering are assumed to have been previously determined and recalculated once per frame at the most. Thus the computational complexity involved in determining the filters has only a limited impact on the total computational complexity and is therefore excluded from the calculation. The computational complexity determination in section 3.4.1 can be used to calculate the complexity involved in applying the SP filter, since a Gaussian filter is used to determine the quality map Q . The computational complexity of the background filter is then $4LN_{Bg}$ for filter kernels of size $L \times L$, where N_{Bg} is the number of pixels in the background.

Method for reduced complexity

It is not necessary to SP filter all background pixels, since the filtering only has an effect in areas containing high frequencies. In figure 3.7 the proposed method for the reduction in computational complexity of the background filtering is presented.

After the quality map has been determined the following steps are taken before the final quality map Q_{Var} is determined.

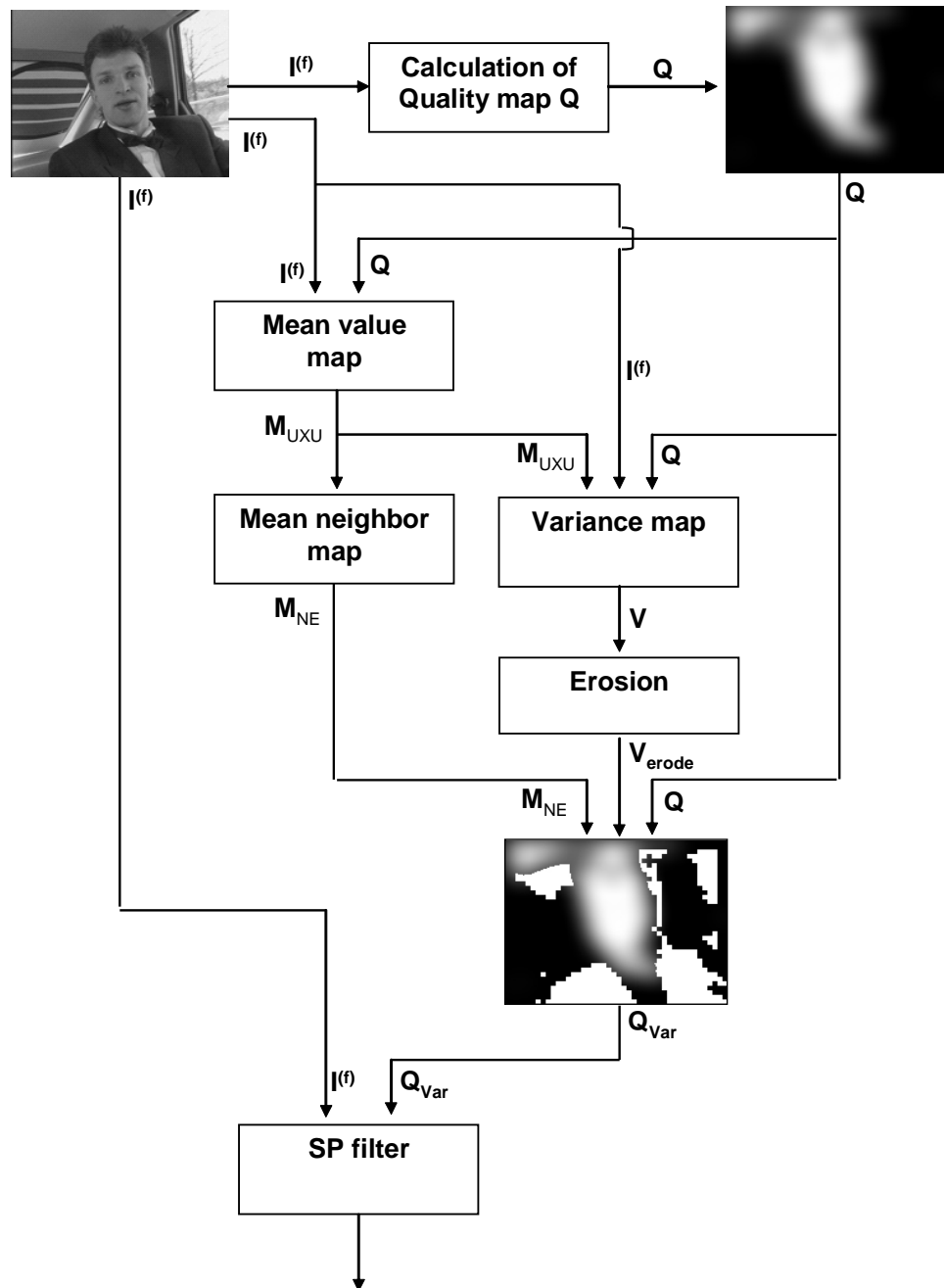


Figure 3.7: The computational complexity of the spatial filter can be reduced by eliminating those parts of the background that would not be noticeably affected by the spatial filter from the filtering. This elimination is based on the variance and mean value within blocks of arbitrary size in the background.

- **Variance map:** The mean value map $\bar{M}_{U \times U}$ and the variance map V containing the average absolute deviation

$$V^{(u,v)} = \frac{1}{U^2} \sum_{m=U \cdot u}^{U(u+1)-1} \sum_{n=U \cdot v}^{U(v+1)-1} |Y^{(f,(m,n))} - \bar{M}_{U \times U}^{(u,v)}|$$

$$\bar{M}_{U \times U}^{(u,v)} = \frac{1}{U^2} \sum_{m=U \cdot u}^{U(u+1)-1} \sum_{n=U \cdot v}^{U(v+1)-1} Y^{(f,(m,n))},$$

is calculated for each $U \times U$ block $B_V^{(u,v)}$ of the intensity image $Y^{(f,(m,n))}$, where $u = \lceil m/U \rceil$, $v = \lceil n/U \rceil$ and $B_V^{(u,v)} \cap ROI = \emptyset$, to find blocks that can be omitted from the filtering. The blocksize $U \times U$ is arbitrary and is therefore not necessarily the same size as a block B used for the DCT or an MB. However, U should be a power of two for it to be proportional to the block sizes used in the block-based hybrid coding.

- **Erosion** The SP filtering with filter kernels of size $L \times L$ (See section 3.2.3) uses the values of pixels outside the $U \times U$ block $B_V^{(u,v)}$ in the calculation of the filtered value of one of the pixels (m, n) within $B_V^{(u,v)}$. A morphological 3×3 erosion filter is applied to ensure that only blocks with neighboring blocks, which contain little detail, are excluded from the spatial filtering.
- **Mean neighbor map** When an edge in the original image is positioned at the border between blocks the difference in mean value between blocks can be substantial even if the variance within both blocks is low. Therefore, the mean value of each block is compared to its four closest neighbors as in

$$\bar{M}_{Ne}^{(u,v)} = |\bar{M}_{U \times U}^{(u,v)} - \frac{1}{4} \bar{M}_{Ne, SUM}^{(u,v)}|$$

$$\bar{M}_{Ne, SUM}^{(u,v)} = \bar{M}_{U \times U}^{(u-1,v)} + \bar{M}_{U \times U}^{(u+1,v)} + \bar{M}_{U \times U}^{(u,v-1)} + \bar{M}_{U \times U}^{(u,v+1)}$$

The maximum distance between a pixel in block (u, v) and a pixel in a neighboring block is approximately twice that between two pixels in the same block. Therefore the maximum allowed mean value difference between two neighboring blocks T_m is defined as twice the lower variance threshold T_v , $T_m = 2T_v$.

The eroded variance map V_{erode} and the neighbor mean map \bar{M}_{Ne} are thereafter combined. The result shows, which $U \times U$ block is able to be excluded in the spatial filter. The resulting modified quality map Q_{var}

$$Q_{Var}^{(m,n)} = \begin{cases} 1 & \text{if } V_{erode}^{(u,v)} \leq T_v, \bar{M}_{Ne}^{(u,v)} \leq 2T_v \\ Q^{(m,n)}, & \text{otherwise} \end{cases}$$

where $u = \lceil m/U \rceil$, $v = \lceil n/U \rceil$, T_v the lower threshold of the variance affecting the spatial filter. The spatial filter in section 3.2.3 can be directly applied to the modified quality map.

The costs of applying this method are.

- Mean value map $\bar{M}_{U \times U}$: U^2 operations per $U \times U$ block
- Variance map V : $2U^2$ operations per $U \times U$ block
- Erosion filter: 9 operations per $U \times U$ block
- Neighbor mean map \bar{M}_{Ne} : 5 operations per $U \times U$ block

Thus the cost of applying the variance detection method in order to create the modified quality map is less than 4 operations per pixel, assuming $U \times U \geq 4 \times 4$. The variance detection method thus results in reduced computational complexity compared to spatial filtering the complete background if

$$4LN_{Bg} > 4L(N_{Bg} - N_{Bg,skip}) + 4N_{Bg}$$

$$\frac{N_{Bg,skip}}{N_{Bg}} > \frac{1}{L}$$

where N_{Bg} is the total number of background pixels and $N_{Bg,skip}$ is the number of background pixels excluded from the filtering. A reduction in computational complexity is thus achieved if a least $1/L$ of the background pixels are excluded from the filtering.

3.5 Experimental setup

The performance of the SP filters were evaluated using the QCIF sequences Carphone and Foreman for 25 fps. The Foreman sequence has been edited so that it only include the first 233 frames in order to ensure that faces are included in all frames. The QCIF size corresponds to frame sizes of $M \times N = 176 \times 144$ and was chosen due to its similarity to the screen sizes of the 3G mobile phones.

The experiment is performed using the following steps:

- ROI Detection: The sequences contain talking heads and thus the ROI is determined using skin-color detection in order to detect the face (See section 2.1.2). The parametric model is presented in [32] and is briefly described in appendix A. It was chosen based on its simplicity and was applied with experimentally

determined thresholds at 30 % (Carphone) and 32 % (Foreman) of the maximum value of the parameterized model in order to extract the binary detection map B_{DM} for each frame. This model has a limited accuracy and is not robust to changes in illumination. However in the chosen sequences, problematic illumination is not present and the face is detected in all frames along with skin-color which is similar to background in some cases.

- **Quality map:** The B_{DM} from the previous step was used as the basis for the quality map. The filters used to create the quality map have filter kernels of size $J \times J = 35 \times 35$, and when a Gaussian filter is used it has a variance $\sigma^2 = (J/4)^2$. This kernel size is sufficiently large to ensure that holes caused by the eyes, mouth and misdetection are filled and small detected areas are excluded. The ROI border is defined using the parameter $A = 1/3$ (See section 3.2.2) giving an average ROI size of 32% (Carphone) and 25% (Foreman), as a percentage of the frame.
- **SP filter:** In the tests the numbers of filters used for the SP filter are limited to 5, 7 and 9, since a higher number would increase the complexity of the algorithm. As given in section 3.2.3 the variance of the filters is calculated based on the maximum variance σ_1^2 . Tests using only one low pass filter with a maximum variance σ_1^2 for the complete background, including the transition area, were also performed as a point of reference for the multiple filter case. Tests on SP filtering based on the alternative version of the quality map Q , which was modified based on variance and mean, were applied using 9 filters with a maximum variance of $\sigma_1^2 = 5^2$. These parameters were chosen based on the values appearing to give the best performance in the tests concerning $PSNR_{ROI, Avg}$ and the score in subjective tests, when the quality map Q was applied. Due to the limited test material it is a possibility that these parameters may not be the optimal parameters to use in the general case.
- **Codec:** The resulting sequences were encoded and decoded using two different codecs belonging to separate standards in order to test the standard independence of the filtering. The MPEG-2 codec, ffmpeg [61], and the H.264 codec, JM 10.1 [62] for the High Profile, were used in the tests.
- **Objective performance evaluation:** As performance measures we determine the average PSNR of the ROI and of the border of the intensity component $Y^{(f, (m, n))}$ for frame f and the total number of frames F

$$PSNR_{ROI, Avg} = \frac{1}{F} \sum_{f=1}^F 10 \log_{10} \frac{255^2}{MSE_{ROI}^{(f)}}$$

	Bit rate	Carphone	Foreman
H.264	max	55 kbps	60 kbps
H.264	min	21 kbps	24 kbps
MPEG-2	max	160 kbps	220 kbps
MPEG-2	min	70 kbps	95 kbps

Table 3.1: The bit rates used in the tests where the max bit rate and min bit rates corresponds to a $PSNR_{Avg}$ of 34 dB and 30 dB, respectively.

$$PSNR_{Border,Avg} = \frac{1}{F} \sum_{f=1}^F 10 \log_{10} \frac{255^2}{MSE_{Border}^{(f)}}$$

$$MSE_C^{(f)} = E_C \{ (Y^{(f,(m,n))} - \hat{Y}^{(f,(m,n))})^2 \}$$

which are based on the average PSNR defined in section 2.4.1. In addition the $PSNR_{ROI}$ per frame is also determined. The H.264 gives a much more efficient compression than that for the MPEG-2 codec which results in a much higher PSNR for the same bit rate. The information in each of the two sequences demands different bit rates in order to achieve the same PSNR. The target bit rates for each original sequence and codec are therefore determined by choosing the target bit rate that gives a certain $PSNR_{Avg}$ in the original sequence. In these tests a max target bit rate and a min target bit rate were applied, that provides an approximate $PSNR_{Avg}$ of 34 dB and 30 dB, respectively for the original sequences. The bit rates are presented in table 3.1.

- Subjective performance evaluation: The perceived quality is not completely represented by the $PSNR_{ROI,Avg}$ measure since it does not consider the impact of introducing artifacts to the background by filtering. Therefore a subjective test was performed in order to verify the assumption that the coding method increases the perceived quality.

A subjective test was performed on pairwise images in order to verify the effect on perceived quality when using several, rather than merely one, filters. Each test subject was then asked to assess the quality of one of the images compared to the other. The assessment was made by marking whether the test image was

better, the same or worse than the reference image. The images were taken from spatially filtered carphone sequences filtered by means of 5-9 filters and encoded using MPEG-2 at 30 fps with bit rates of 100 kbps and 150 kbps. These were compared for each test pair to an image from the corresponding video sequence where only one filter had been applied and the sequence was encoded in the same manner as that for the 5-9 filtered sequence. The order in which the pairs were tested was randomized in addition to which was presented as the reference frame in each pair in order to avoid effects from fatigue and that some test subjects will favor a positive choice before a negative.

3.6 Experimental results

In this section the results of tests measuring the objective quality in the form of $PSNR_{ROI,Avg}$ and $PSNR_{Border,Avg}$ and subjective quality tests are presented and analysed. It can be assumed that the main characteristics of the experimental results could be transferred to sequences with larger frame size. This assumption is based on the fact that generally an increased frame size in both the ROI and background will give a proportional increase in detail. Thus more detail is available for removal in the background at the same time as the demand for bits, to ensure good quality in within the ROI, increases. The relationship between the size of the ROI and size of the background will remain the same for all frame sizes assuming that the same video sequence is viewed.

3.6.1 SP filtering using several Gaussian filters

In figures 3.8 and 3.9 it can be seen that by applying one or several gaussian filters and encoding the sequence using H.264, the $PSNR_{ROI,Avg}$ is increased by at least 1.9 dB for the max bit rate and 1.0 dB for the min bit rate for $\sigma_1 > 2.5$. The use of only one filter causes border effects for larger σ_1 , which is an explanation for the moderate increase of approximately 0.1 dB for 5-9 filters compared to that when using only one filter which removes more information in the background and, thus, in principle should have a better $PSNR_{ROI,AVG}$. However, the complete effect on perceptual quality of these border effects can not be detected by the PSNR measure as it only consider the error per pixel and not in which context this pixel is found. In the case of only one filter better results are shown for $\sigma_1 < 2.5$, where the probability of border effects is low. There is no noticable difference between using 5, 7 or 9 filters. A similar result can be seen in figures 3.10 and 3.11 when an MPEG-2 codec has been applied. The reason why the resulting curves are less smooth than for the H.264 case is that the chosen MPEG-2 encoder is less able to adapt to a particular target

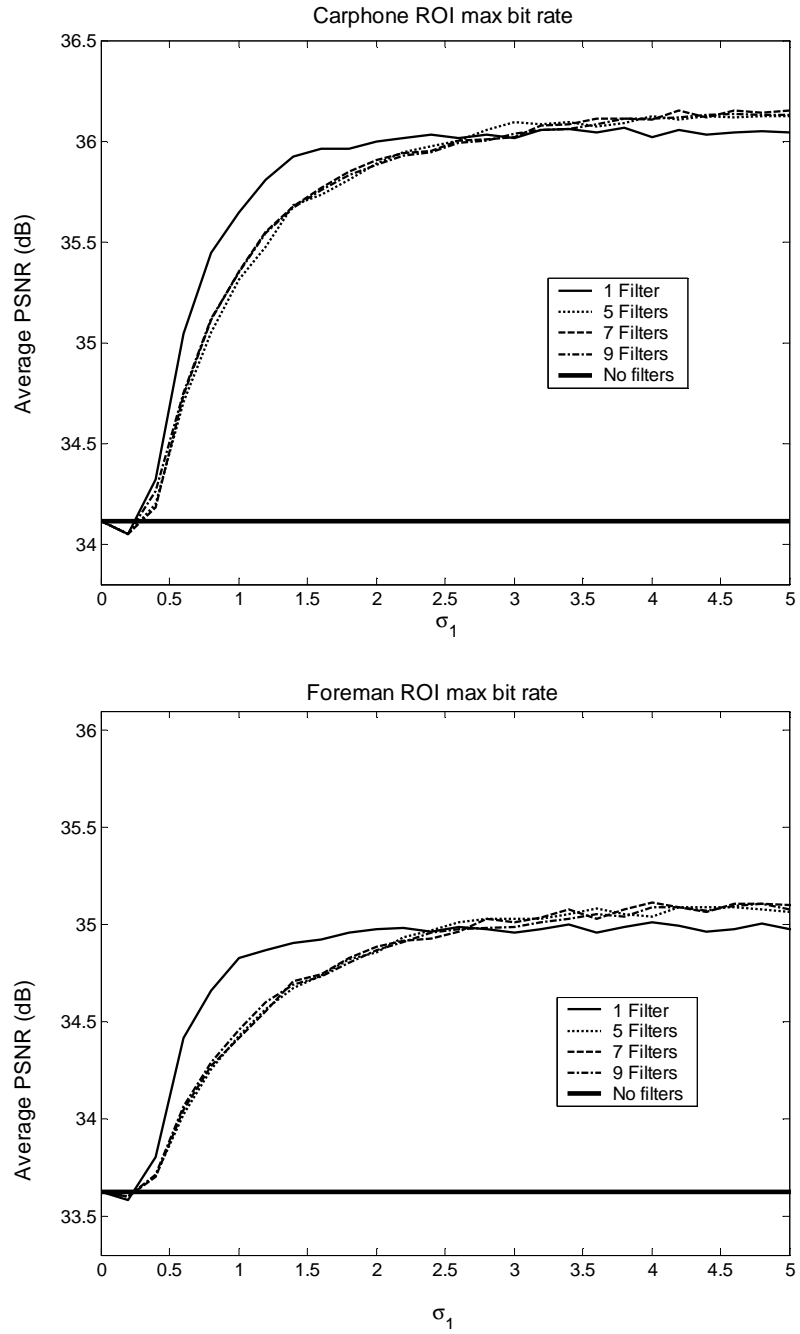


Figure 3.8: The $PSNR_{ROI,Avg}$ for different values of σ_1 and different number of gaussian filters are presented for h.264 encoded with max bit rate for the carphone sequence (top) and the foreman sequence (bottom).

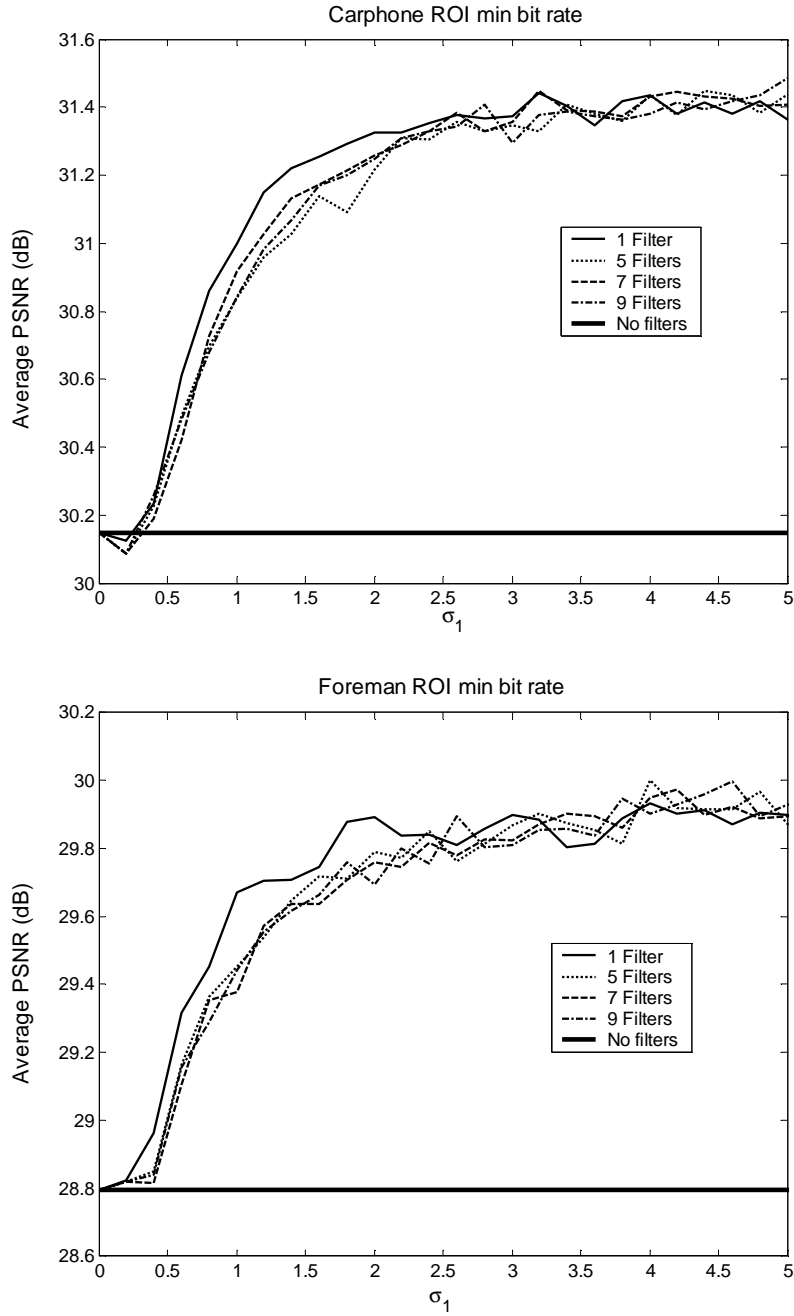


Figure 3.9: The $PSNR_{ROI, Avg}$ for different values of σ_1 and different number of gaussian filters are presented for h.264 encoded with min bit rate for the carphone sequence (top) and the foreman sequence (bottom).

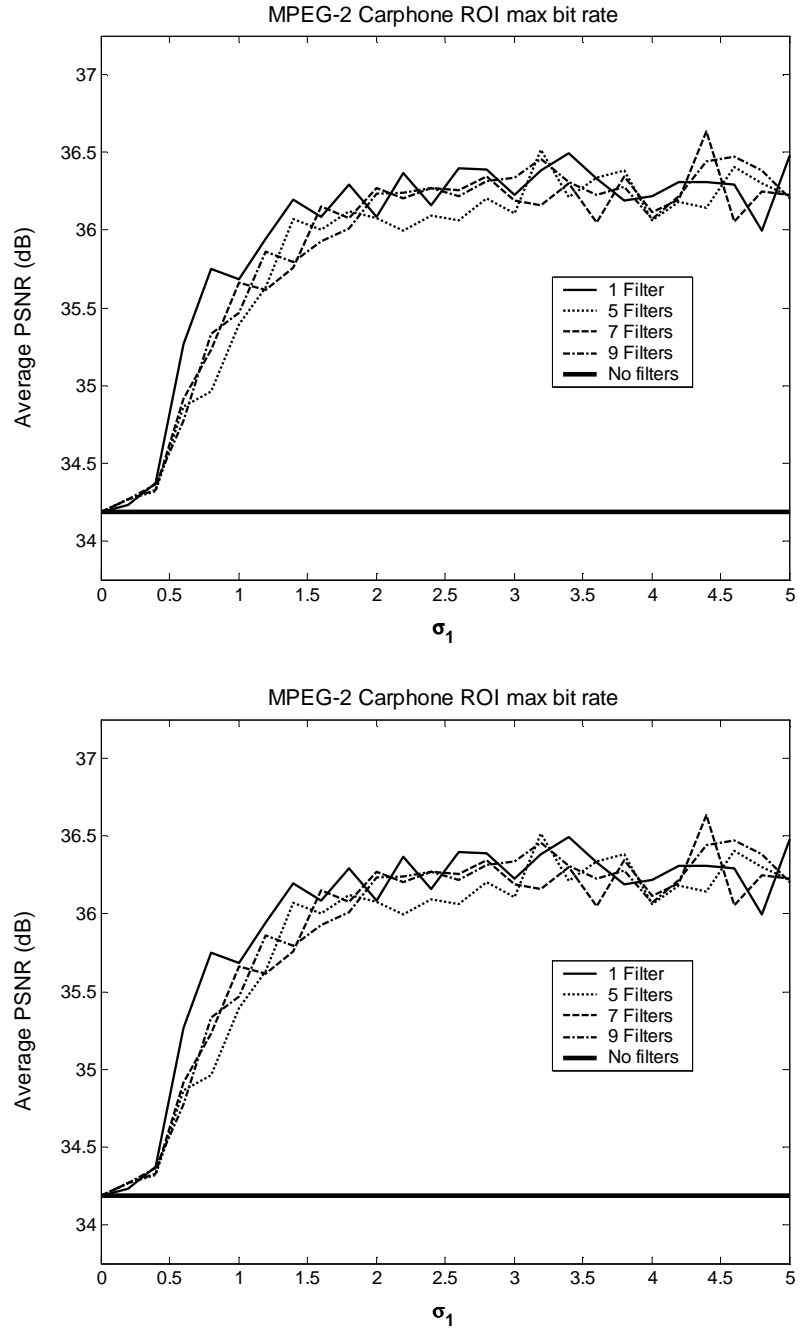


Figure 3.10: The $PSNR_{ROI,Avg}$ for different values of σ_1 and different number of gaussian filters are presented for the with max bit rate MPEG-2 encoded carphone sequence (top) and the foreman sequence (bottom).

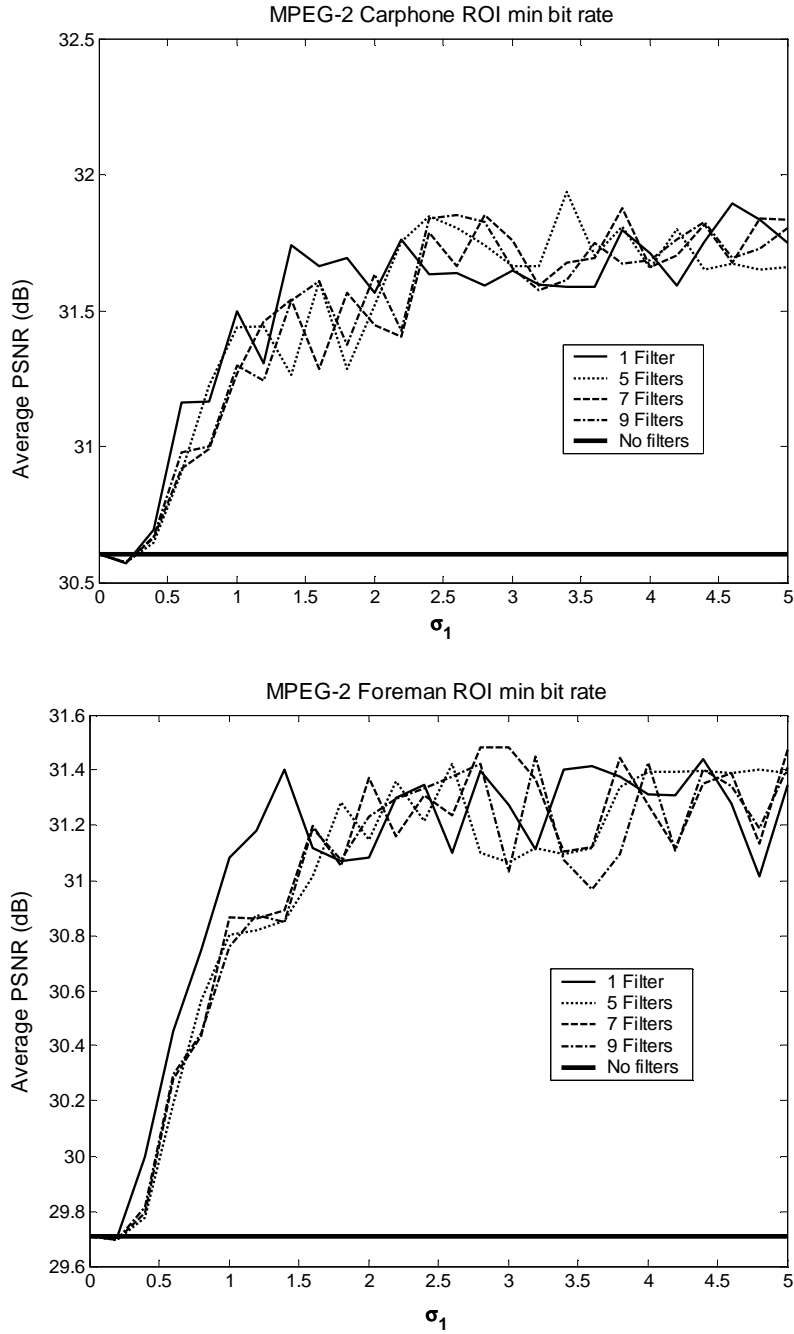


Figure 3.11: The $PSNR_{ROI, Avg}$ for different values of σ_1 and different number of gaussian filters are presented for the with min bit rate MPEG-2 encoded carphone sequence (top) and the foreman sequence (bottom).

bitrate than the H.264 codec. Thus the variation in final bit rates is much larger in the MPEG-2 case than for the H.264 codec.

The quality of the border area shows an increase, when using 5-9 filters instead of one filter of $PSNR_{Border,Avg}$ of 0.4-1.5 dB for the Carphone sequence and 0.2-1.2 dB for the Foreman sequence considering $\sigma_1 > 2.5$ for H.264. Plots of the $PSNR_{Border,Avg}$ for different σ_1 and different numbers of Gaussian filters can be found in figures B.1 and B.2 in appendix B. This increase is due to the gradual quality degradation, which does not remove as much details as when only one filter is applied. In principle less quality in the background $PSNR_{Border,Avg}$ implies less bits to re-allocate to the ROI and thus a lower $PSNR_{ROI,Avg}$. However the results show that this is not the case when using 5-9 filters for $\sigma_1 > 2.5$ in comparison to the case involving only one filter. The $PSNR_{ROI,Avg}$ is even marginally improved by using 5-9 filters even though the $PSNR_{Border,Avg}$ is higher for 5-9 filters than for only one filter. The MPEG-2 encoding gives a similar result but with a smaller increase of $PSNR_{Border,Avg}$ for 5-9 filter than for H.264.

3.6.2 Reduction of computational complexity

In section 3.4.2 two methods for reducing computational complexity were proposed. The first is intended to reduce the number of operations when creating the quality map Q from the binary detection map B_{DM} by applying the mean value filter $\bar{M}_{J \times J}$ in equation (3.11) instead of the Gaussian filter G in equation (3.5). Tests showed that applying $\bar{M}_{J \times J}$ instead of G gives a maximum variation of the $PSNR_{ROI,Avg}$ of ± 0.2 dB, which is not considered to be of significance.

The second approach aimed at reducing the number of spatially filtered background pixels by using variance and mean value measures. This method resulted in a reduction in $PSNR_{ROI,Avg}$ as the variance threshold T_v increases, which can be observed in figure 3.12. The case involving the use of the original quality map is represented by $T_v = 0$. The corresponding reduction in percentage of filtered pixels in the background is presented in figure 3.13. It is assumed that a decrease of 0.2 dB for the max target bit rate and 0.1 dB for the min target bit rate cannot be detected by the human eye. In figure 3.12 the variance threshold, where the decrease in $PSNR_{ROI,Avg}$ is equal to 0.2 dB for max target bit rate and 0.1 dB for min target bit rate is marked by a line for each $U \times U$ size. The same threshold is also marked with a line in figure 3.13 indicating the percentage of pixels requiring to be filtered in order to ensure no visual reduction in $PSNR_{ROI,Avg}$. Thus only 66% of the background require to be filtered in these sequences assuming $U \times U = 4 \times 4$. It can also be seen that a better performance is provided by the block size $U \times U = 4 \times 4$ than that for $U \times U = 8 \times 8$. In section 3.4.2 it was shown that if more than $\frac{1}{L}$ ($L \times L =$ the size of

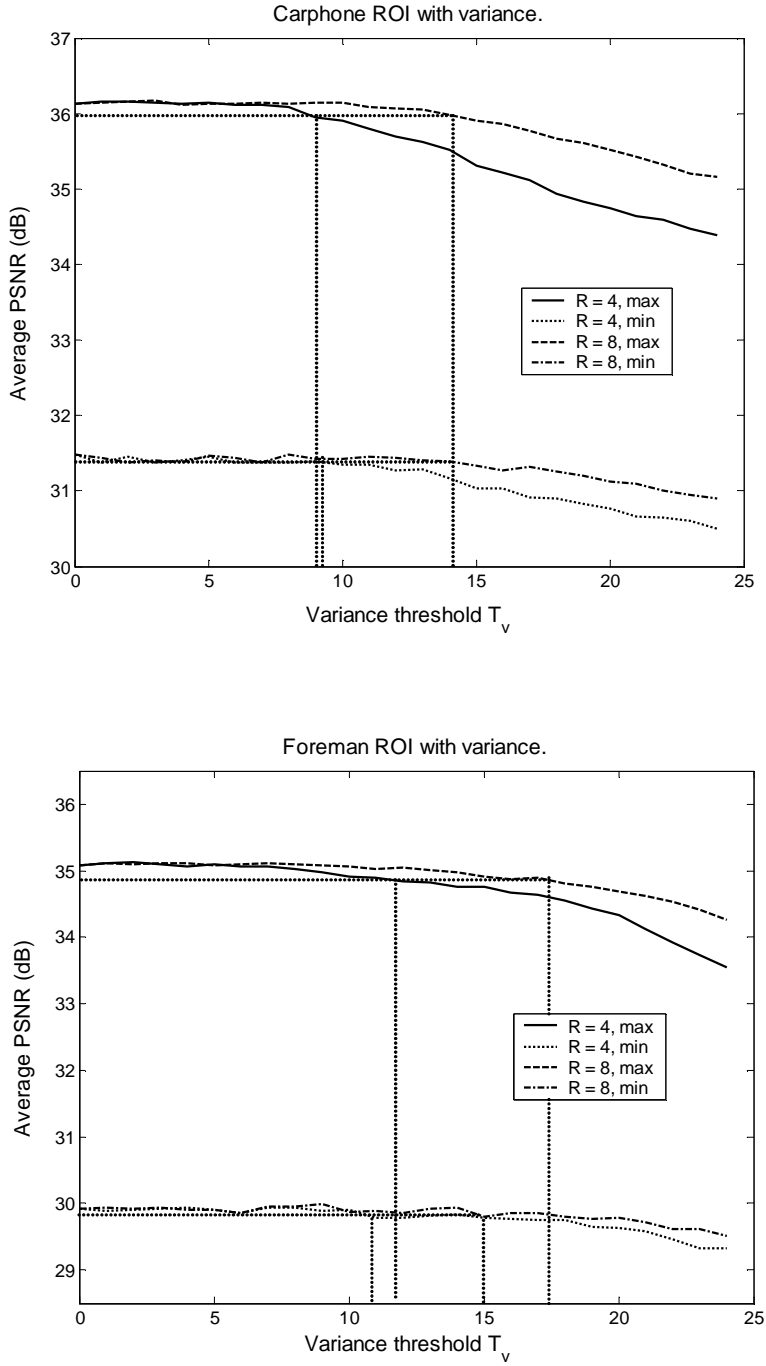


Figure 3.12: The $PNR_{ROI, Avg}$ for different variance thresholds T_v are presented in this figure for the carphone sequence (top) and the foreman sequence (bottom). A variance threshold of $T_v = 0$ corresponds filtering all background pixel.

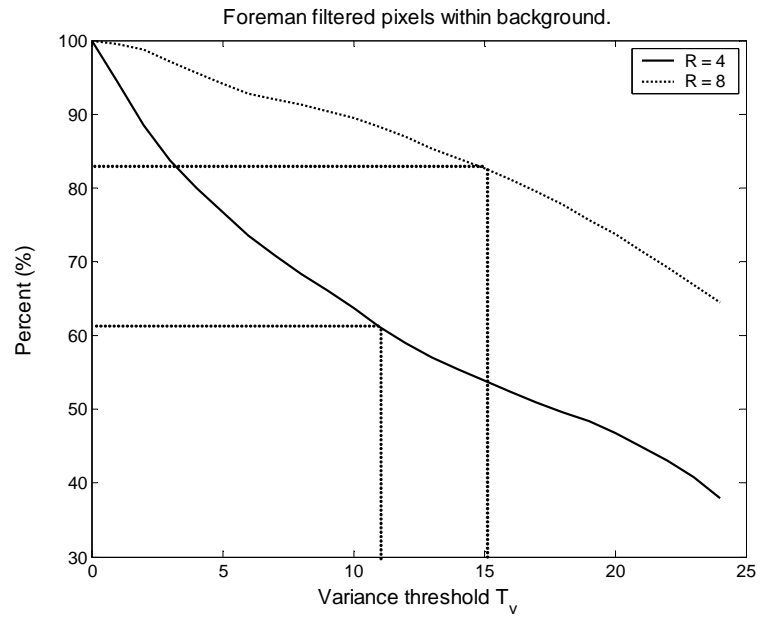
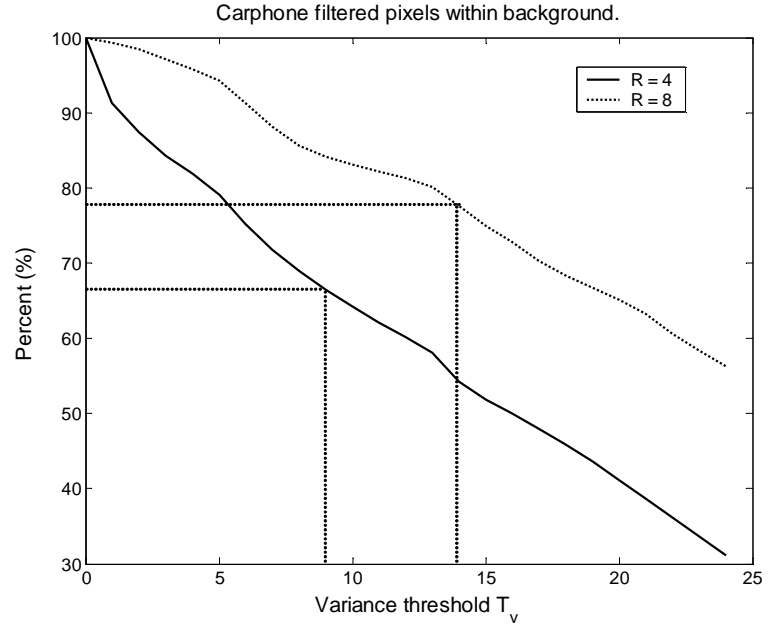


Figure 3.13: The percentage of spatially filtered background pixels for different variance thresholds T_v are presented in this figure for the carphone sequence (left) and the foreman sequence (right).

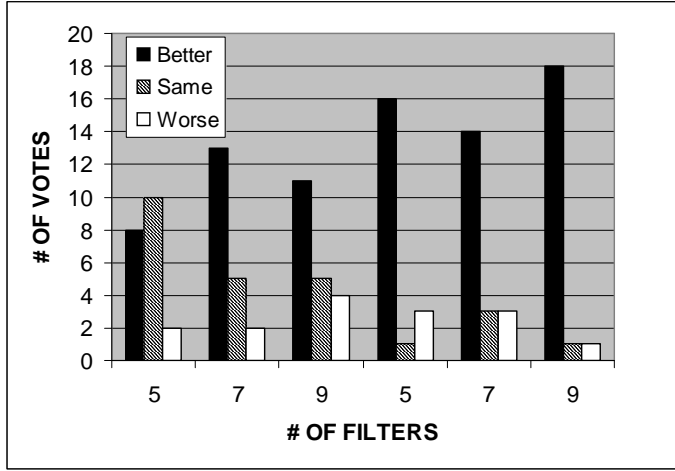


Figure 3.14: The result of the subjective test where the testsubject were asked to assess the quality of pairwise presented images, where one is spatially filtered with 5-9 gaussian filters and compared to an image filtered with one gaussian filter.

the filter kernels in the spatial filter) background pixels were skipped in the filtering, this method reduces the computational complexity. Thus by applying the approach in section 3.4.2 to exclude blocks with low variance, then the computational complexity is reduced since $0.34 > \frac{1}{L} = \frac{1}{5}$.

3.6.3 Subjective tests

In the first subjective tests applied to still images a majority of the test subjects found that applying the variable filter approach using 5-9 filters gave a better quality than when using only 1 filter (See figure 3.14). The tests also showed that the improvement in perceptual quality was more noticeable for 150 kbps than for 100 kbps. In section 3.6.1 it was shown that the $PSNR_{ROI, Avg}$ is not noticeably affected by the choice of 5, 7 or 9 filters, but the result in figure 3.14 shows that the likelihood of noticeable boundary effects are higher, the fewer the filters used.

3.7 Chapter summary

A previous idea relating to SP pre-processing filters has been improved by introducing a gradual quality transition from the background to the ROI by applying 5-9 Gaussian filters with different variances. The decision relating to which filter to ap-

ply is based on values in the quality map, which is extracted by lowpass filtering the binary detection map after ROI detection. The results of the theoretical analysis and the quantitative tests are summarized in table 3.2.

The contribution of the author to this chapter includes:

- The determination of the quality map and its use to control the variance of a limited set of Gaussian filters instead of one filter in [11]. The idea of using several filters was partially introduced in [18] where a pyramid of Gaussian filters was applied to create a gradual quality transition based on the distance to a point. This is extended to include complete ROI's by determining the distance to the border of a region instead of to single point. The position of the ROI and the distance to the ROI border can be extracted from the quality map.
- The effect of the spatial filter on the calculation of rate and distortion, which is a part of the automatic determination of coding parameters to achieve a target bit rate, was analysed based on the general hybrid block-based encoder and standards.
- A computational complexity analysis together with two methods to reduce the computational complexity.
- Test results from both objective measures, such as the PSNR, and subjective tests and an analysis of these.

Coding efficiency of the background.	Less bits allocated to DCT coefficients, prediction error and motion vectors.
Re-allocation from background to ROI.	The bits released by the SP filter are reallocated mostly to the ROI, where the most DCT components are present or the prediction error is the largest.
Computational complexity	<p>Detection of the ROI algorithm: From MN operations per frame or more depending on the detection algorithm ($M \times N$ = number of pixels per frame).</p> <p>Quality map: Gaussian filter kernel: $4JMN$ operations per frame for a $J \times J$ size filter kernel. Mean value: $2JMN$ operations per frame.</p> <p>SP filter with $L \times L$ filter kernels: $4LN_{Bg}$ operations per frame, but with the computational complexity reduction using variance and mean this reduces to $4L(N_{Bg} - N_{Bg,skip}) + 4N_{Bg}$. N_{Bg} = the number pixels in the background $N_{Bg,skip}$ = the number of pixels skipped in the filtering</p>
PSNR	<p>The SP filtering increases $PSNR_{ROI,Avg}$ by at least 1.9 dB and 1.0 dB, for max and min bit rate, respectively, when coding with the H.264 codec.</p> <p>Applying 5-9 filters give a moderate increase in $PSNR_{ROI,Avg}$ and an increase of 0.2 - 1.5 dB in $PSNR_{Border,Avg}$ for $\sigma_1 > 2.5$ for the H.264 codec.</p> <p>MPEG-2 gives a similar result as the H.264 codec.</p>
Subjective tests	<p>5,7 or 9 filters are experienced as better in most cases.</p> <p>The test score was better the larger the number of filters.</p>

Table 3.2: A summary of the results of the theoretical analysis and the qualitative tests performed on the SP filter

Chapter 4

Temporal filtering

The proposed temporal filter presented in this chapter extends the approach in [25] in order to provide independence from the codec. This is achieved by controlling the blocks which should be skipped by the pre-filtering. Bilinear interpolation of the area close to the ROI border is also included so as to minimize artifacts due to large movement of the ROI.

The two versions of the temporal filter described in figures 4.1 and 4.2 are presented in detail in section 4.1. The performance of the filter has been evaluated using both a theoretical analysis as well as qualitative tests. The theoretical analysis includes an analysis of the effects of the TP filter on the rate distortion optimization in section 4.2 and the computational complexity in section 4.3. The results of the quantitative tests are presented in section 4.5 using bit rate and PSNR as performance measures.

4.1 Temporal filter

The position of the ROI, background and the transition area of the ROI are determined from the ROI detection and the resulting quality map, which is the same as in the spatial case (See section 3.2.2). This information is used to control the temporal filter in this section.

The proposed algorithm described in figure 4.1 performs a temporal filtering (TP) in order to achieve a reduction in the frame rate by a factor two in the background without coding and transmitting the sequence in separate parts, which would have to be synthesized in the decoder. The TP filter is performed on blocks $B_{TP}(p, q)$ instead of on pixels (m, n) as for the SP filter. This is mainly because the aim of

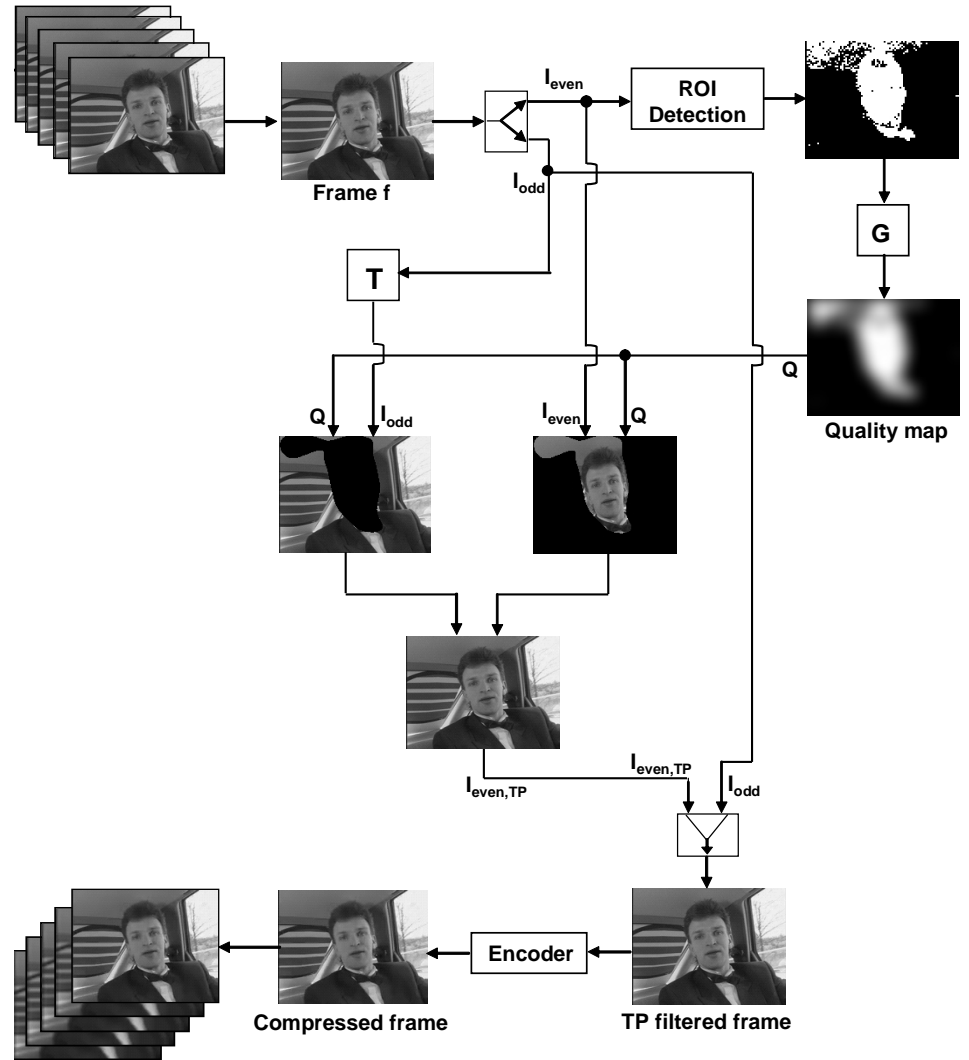


Figure 4.1: The first version of the temporal filter including the ROI detection, calculation of the quality map and filtering of the background.

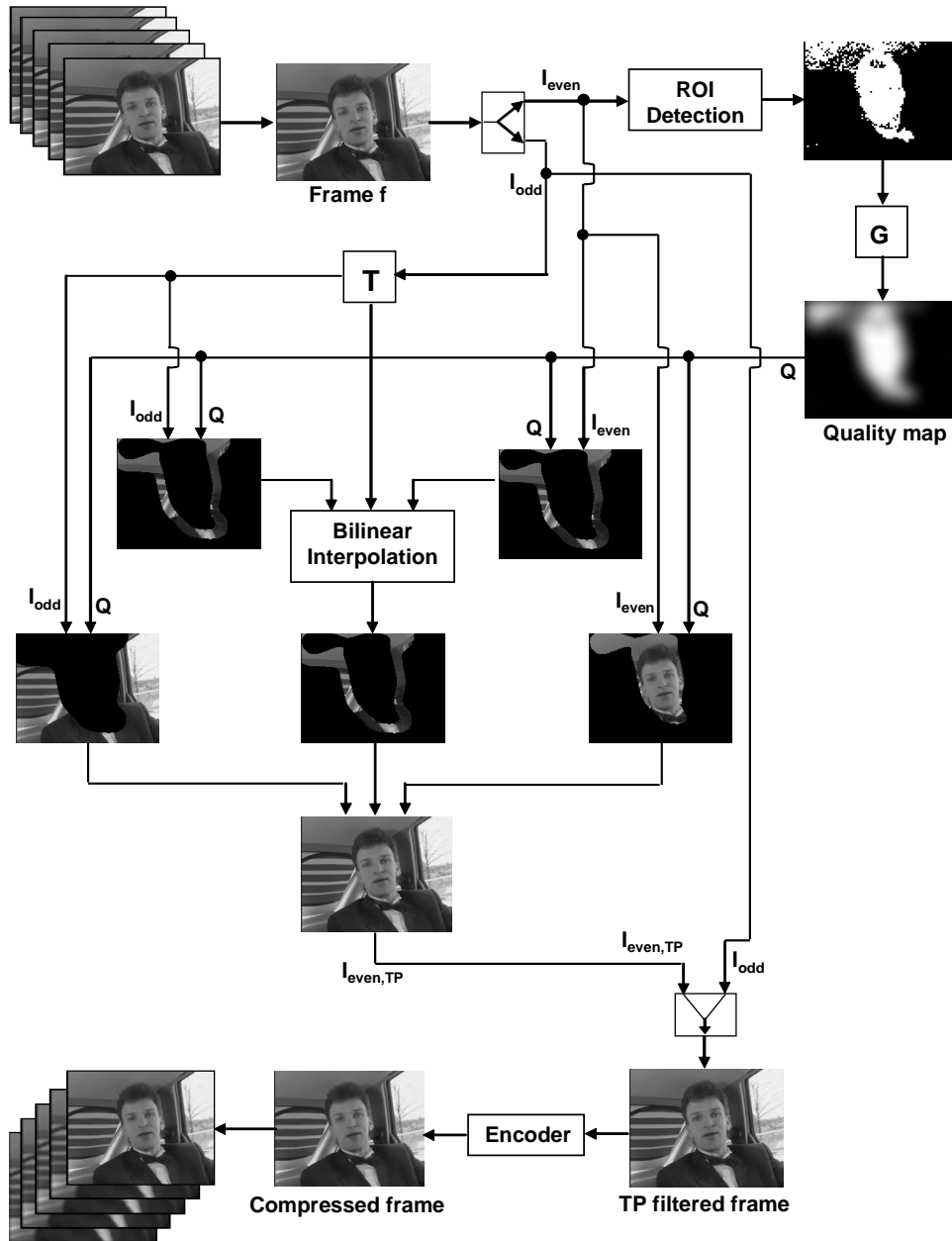


Figure 4.2: The second version temporal filter including the ROI detection, calculation of the quality map and filtering of the background with bilinear interpolation.

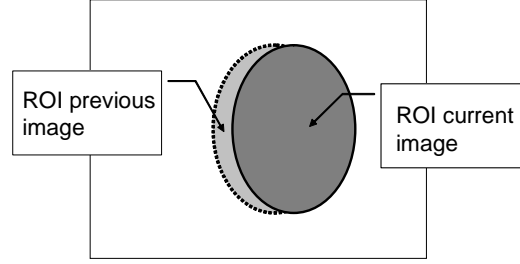


Figure 4.3: If there is large movement of the ROI border some of the background could be predicted from the ROI of the previous frame (left) or vice versa. This causes border effects (right).

the TP filter is to effect the prediction of a block-based hybrid encoder, which is determined on a MB basis as described in section 3.1. However, in H.264 [7] smaller block sizes are allowed in the prediction of an MB and therefore the size of the blocks in the TP filter are assumed to be 8×8 . The values for the resulting filtered frame are determined by combining the values from even frames \mathbf{I}_{even} with the previous odd frame \mathbf{I}_{odd} , allowing only the ROI to contain new information. Thus for every block $B_{TP}(p, q)$ in \mathbf{I}_{even} the corresponding block in the filtered frame becomes

$$\mathbf{I}_{TP}^{(p,q)} = \begin{cases} \mathbf{I}_{even}^{(p,q)}, & \text{if } B_{TP}(p, q) \cap ROI \neq \emptyset \\ \mathbf{I}_{odd}^{(p,q)}, & \text{otherwise} \end{cases} \quad (4.1)$$

$$p = \lceil m/8 \rceil \text{ and } q = \lceil n/8 \rceil$$

The border between the ROI and background is not stationary between frames. This leads to problems similar to those when combining layers in an MPEG-4 decoder [47]. Large movements of the ROI from frame to frame will cause the background in \mathbf{I}_{even} to be assigned values from the ROI in the previous \mathbf{I}_{odd} (See figure 4.3). Corresponding artifacts also occur if the ROI of \mathbf{I}_{even} covers the background of the previous \mathbf{I}_{odd} .

In the proposed TP filter in figure 4.2 these artifacts are compensated for by applying a gradual transition of quality from the ROI to the background. The classification as to whether a block belongs to the ROI, transition region or background depends on the maximum value of the quality map within that block. An example is given in figure 4.4. Thus the ROI includes all blocks where $B_{TP}(p, q) \cap ROI \neq \emptyset$ and the transition region is defined as all blocks $B_{TP}(p, q)$ where $A_{Bg} \leq Q^{(p,q)} < A_{ROI}$ and $Q^{(p,q)} = \max(Q^{(m,n)})$ for $(m, n) \in B_{TP}(p, q)$. The threshold giving the position of the ROI border $A_{ROI} = 1/3$ and its determination can be found in section 3.2.2.

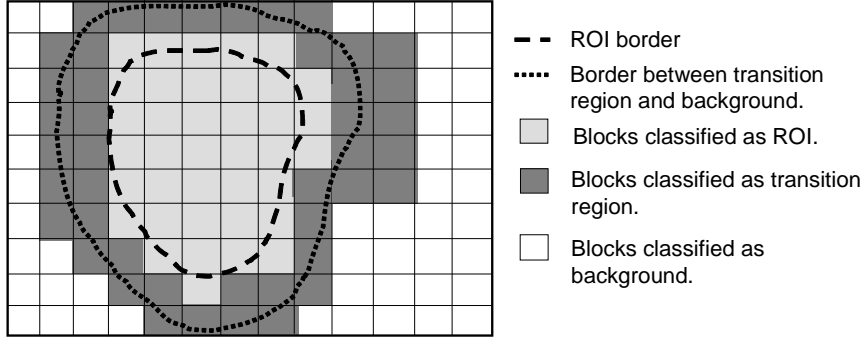


Figure 4.4: The classification of blocks to ROI, transition region and background from a pixel-based quality map Q .

The lower threshold of the transition area A_{Bg} is chosen such that it is close to 0 to ensure as large a transition region as possible, thus $A_{Bg} = 0.01$. The addition of interpolation of this region results in a blurred transition region without sharp artifacts, where bilinear interpolation is chosen based on its simplicity. It gives the following modification to the TP filter. For every block $B_{TP}(p, q)$ in \mathbf{I}_{even} , the filtered frame is

$$\mathbf{I}_{TP}^{(p,q)} = \begin{cases} \mathbf{I}_{even}^{(p,q)}, & \text{if } B_{TP}(p, q) \cap ROI \neq \emptyset \\ f_{bl} = (\mathbf{I}_{even}^{(p,q)}, \mathbf{I}_{odd}^{(p,q)}), & \text{if } A_{Bg} \leq Q^{(p,q)} < A_{ROI} \\ \mathbf{I}_{odd}^{(p,q)}, & \text{otherwise} \end{cases} \quad (4.2)$$

where $f_{bl}(\mathbf{I}_A, \mathbf{I}_B) = \alpha \mathbf{I}_A^{(m,n)} + (1-\alpha) \mathbf{I}_B^{(m,n)}$ is bilinear interpolation such that for each $(m, n) \in B_{TP}(p, q)$ in the transition region $\alpha = A_{ROI} Q^{(m,n)}$.

4.2 Rate-Distortion of TP filtered video

In this section the encoding of the TP filtered video sequences will be addressed and compared to encoding the original video sequences. The analysis is based on the definitions in section 3.1. In this analysis the intra-coded frames are addressed in section 4.2.1 followed by the inter-coded frames in 4.2.2.

4.2.1 Intra-coded frames

The difference of the spatial filter addressed in 3.3.1 compared to the temporal filter is that it does not remove frequency components from any part of the spectrum and

therefore has no effect on the compression of the background in intra-coded frames. However, the temporal filter without bilinear interpolation introduces new edges at the border for large movements of the ROI from one frame to the other. This gives additional high frequency components to encode, but only at the ROI border.

4.2.2 Inter-coded frames

The removal of changes in the background by the temporal filter gives a prediction error for even frames of,

$$E\{|\mathbf{I}_{Bg,TP}^{(f,(m,n))} - \mathbf{I}_{Bg,TP}^{(f-1,(m,n)+\bar{d}_{MV})}|^2\} = 0.$$

for $\bar{d}_{MV} = (0,0)$, since $\mathbf{I}_{Bg,TP}^{(f,(m,n))} = \mathbf{I}_{Bg,TP}^{(f-1,(m,n))}$ in this case. Thus the background blocks can be skipped in the encoding thus for even frames $R_{Bg,TP}^{(f)} = R_{Bg,TP,OH}^{(f)}$.

The backgrounds of the odd frames, however, are predicted using data from the previous odd frame which means that the motion vector $\bar{d}_{MV,TP}$ is chosen by minimizing

$$E\{|\mathbf{I}_{Bg,TP}^{(f,(m,n))} - \mathbf{I}_{Bg,TP}^{((f-1),(m,n)+\bar{d}_{MV,TP})}|^2\} \approx E\{|\mathbf{I}_{Bg}^{(f,(m,n))} - \mathbf{I}_{Bg}^{(f-2),(m,n)+\bar{d}_{MV}}|^2\},$$

since $\bar{d}_{MV,TP} = \bar{d}_{MV}^{(f-1)} + \bar{d}_{MV}^{(f-2)}$ as described in figure 4.5 where the background of frame $f-1$ is equal to the background of frame $f-2$. However in the case involving a large change between two frames, either due to noise or scene changes, the best match in frame $f-2$ in the original case might not match the best match $f-1$ in the TP filtered case even though the the background contains the same information. On the other hand, it can be assumed that the chosen best match costs less bits to encode than that given in figure 4.5 and therefore this is disregarded in the analysis. In addition intra-coding of the MB or prediction from ROI MBs might cost less bits, than predicting motion vectors from background MBs, and therefore be chosen. The analysis is simplified by assuming that only background blocks are used in the prediction, which gives a higher number of bits allocated to motion vectors of the filtered frame $R_{Bg,TP,MV}^{(f)}$ than in reality. Under this assumption, the resulting motion vector $\bar{d}_{MV,TP}$ is always shorter than the sum of the two motion vectors in the unfiltered case, $\|\bar{d}_{MV,TP}\| \leq \|\bar{d}_{MV}^{(f-1)}\| + \|\bar{d}_{MV}^{(f-2)}\|$.

The number of bits allocated to $\bar{d}_{MV,TP}$, which is twice as large as $\bar{d}_{MV}^{(f-1)}$, depends on the encoding algorithm. In the H.261 [4], MPEG-2 [3] and H.263 [5] standards a table of variable length codes (vlc) is applied. One codeword per codeword length is applied for a motion vector with components of length below 2.5 for H.263 and below 5 for H.261 and MPEG-2. Thereafter the number of codewords per codeword length increases.

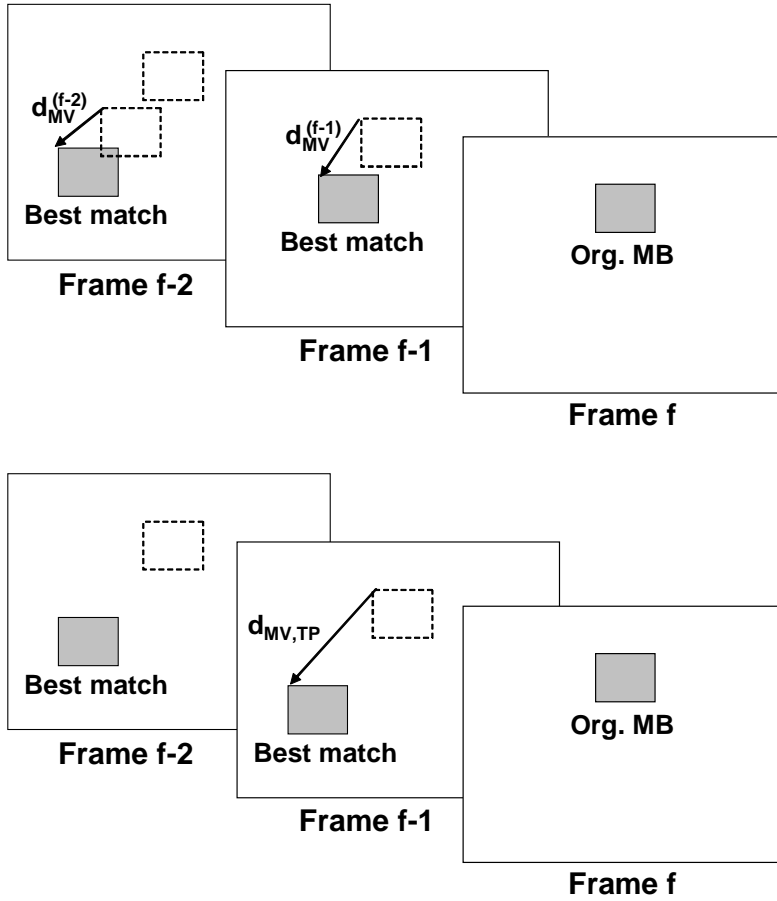


Figure 4.5: In the prediction of a particular MB in frame f the best match in the previous frame $f - 1$ is determined. The best match is the block of same size for which the motion vector $\vec{d}_{MV}^{(f-1)}$ and the resulting prediction error of this block and the MB in frame f costs the least bits to encode. The prediction of an MB in frame f and the prediction of the used MB in frame $f - 1$ is shown for the original sequence (top) and the temporally filtered sequence (bottom). It can be seen that the motion vector of the best match in $f - 1$ in the temporally filtered sequence equals that sum of the two motion vectors in the original sequence. Thus $\vec{d}_{MV,TP} = \vec{d}_{MV}^{(f-1)} + \vec{d}_{MV}^{(f-2)}$.

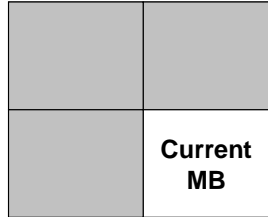


Figure 4.6: For each MB a predicted motion vector determined from the previous MBs, which are grey in the figure, are subtracted from the motion vector given in the motion estimation and the resulting motion vector difference is transmitted instead of the original motion vector.

In the Context-based Adaptive Binary Arithmetic Coding (CABAC) used in H.264 [7] the motion vectors of the neighboring MBs (See figure 4.6) are used to predict the motion vector of the current MB. The difference between the actual motion vector of the current MB and the predicted motion vector, the motion vector difference (MVD), are then encoded. CABAC applies adaptive coding for the shorter MVDs (< 9) by adapting the codeword lengths to the probability of a particular MVD length. The MVD lengths exceeding 9 is encoded using exponential golomb with exponential growth of the number of code words per code word length.

The length of each MVD depends on how similar it is to the surrounding motion vectors. Under the assumption stated earlier in this section and, in particular, that the movement between frames does not vary rapidly, the TP filter provides motion vectors which are approximately twice as long as those in the background. Thus the length of each MVD will remain approximately the same, since the surrounding motion vectors have a proportional increase in comparison to one another. However by doubling the motion vector lengths any variations between neighboring motion vectors increases, which makes the prediction of the motion vector from the neighboring motion vectors less accurate and thus gives a larger MVD. Therefore the TP filter gives a slightly larger mean MVD than when encoding the original sequence if the variance in motion vector lengths in the background is large. This increase in mean MVD length still remains smaller than the original mean MVD length.

In the reasoning concerning the codeword lengths used for the codewords in order to describe the length of motion vector components, for the different standards, clearly, a double length motion vector is encoded using less bits than are used for two

motion vectors which have the original length. In addition information concerning the type of predictive encoding of that block is only sent for those MBs, which are not skipped by the encoder in the odd frames. The movement of a block over two frames by one motion vector can therefore be described by less bits than when using two motion vectors of half size, thus

$$R_{Bg,TP,MV}^{(f)} \leq R_{Bg,MV}^{(f)} + R_{Bg,MV}^{(f-1)},$$

where the equality only applies if there is no movement in the background when compared to the previous two frames.

Thus by encoding the filtered sequence with fixed quantization parameters, the prediction error of each odd frame thus satisfies

$$\begin{aligned} & E\{|\mathbf{I}_{Bg,TP}^{(f,(m,n))} - \mathbf{I}_{Bg,TP}^{(f-2,(m,n)+\bar{d}_{MV,TP})}|^2\} \\ & \leq E\{|\mathbf{I}_{Bg}^{(f,(m,n))} - \mathbf{I}_{Bg}^{((f-1),(m,n)+\bar{d}_{MV}^{(f-1)})}|^2\} + E\{|\mathbf{I}_{Bg}^{(f-1,(m,n))} - \mathbf{I}_{Bg}^{((f-1),(m,n)+\bar{d}_{MV}^{(f-2)})}|^2\}, \end{aligned}$$

where $\bar{d}_{MV}^{(f-1)}$ is the motion vector in the original frame f that points to the MB in $f - 1$ whose motion vector $\bar{d}_{MV}^{(f-2)}$ points at the same MB in $f - 2$ as $\bar{d}_{MV,TP}$ in the temporally filtered case as shown in figure 4.5. Thus

$$R_{Bg,TP,PErr}^{(f)} \leq R_{Bg,PErr}^{(f)} + R_{Bg,PErr}^{(f-1)}.$$

Assuming that video, in general, changes very little from frame to frame the equation (4.2.2) will either be equal or close to equal. More bits are allocated to the encoding of prediction errors in the complete frame, since the minimization in equation (3.2) implies that fewer bits are necessary for the encoding of the motion vectors. However, the distortion of a TP filtered video sequence is minimized by applying more bits to the prediction error in both the background and the ROI, since the TP filter does not reduce the prediction error to the same extent as the SP filter. Exclusive reallocation of bits to the ROI requires additional processing, such as SP filtering or controlling the number of DCT components used on the prediction error.

4.3 Computational complexity

The computational complexity is defined as the number of operations, where an operation consists of additions, subtractions, multiplications and divisions. The computational complexity of the detection and quality map is determined and presented in section 3.4.

Only the bilinear interpolation of the transition region requires consideration when determining the computational complexity of the temporal filter. The bilinear

interpolation costs four operations for each considered pixel. Thus the complexity of the temporal filter is $4N_{Tr}$, where N_{Tr} is the number of pixels in the transition area. No calculations are necessary in order to determine the filtered background pixels not included in the transition area.

4.4 Experimental setup

The experimental setup for the steps required before the TP filter can be applied is the same as for the ROI detection and quality map (See section 3.5.). One additional sequence Closeup is added to the tests and all the sequences are tested for 10, 15, and 25 fps. This sequence was filmed by the author and contains a close-up of a talking face with a panning outdoor background. Therefore the background information of this sequence has an high level of motion. The ROI of the Closeup sequence is detected using the same method as the other sequences with a threshold of 32%. This gives an average ROI size of 49% of the frame.

Two versions of the background filtering were tested, method 1, utilizing the background filtering in equation 4.1 and method 2, including the bilinear interpolation as in equation 4.2. The α parameter of the bilinear interpolation is assumed to be the function $\alpha = Q^{(m,n)}/A$. This choice of function was verified as being a good choice in appendix C. The performance was measured using the quality measures, the average ROI PSNR $PSNR_{ROI,avg}$ and average border PSNR $PSNR_{Border,Avg}$. The $PSNR_{Border,Avg}$ gives an indication of the presence of border artifacts and is calculated for the region, where $A_{Bg} \geq Q^{(m,n)} \geq 0.5$. The upper bound of 0.5 is chosen so that all the effects at the border are included when encoding the MB containing the border pixels and possibly also the ROI pixels. Additional tests with a fixed quantization parameter of $Qp = 28$ have been performed in order to show how many bits are necessary to encode the sequence using fixed parameters. The same codecs and bit rates of the sequences as used in section 3.5 were applied to these tests. However the MPEG-2 codec was only applied in one of the PSNR tests so as give an indication whether its bit-allocation differs from the bit-allocation in the H.264 codec.

4.5 Experimental results

In this section the test results of three types of tests are presented. The first set being those for the bit rate of the encoder given a fixed quantization parameter, which are then followed by the $PSNR_{ROI,avg}$ and $PSNR_{Bg,avg}$ together with the $PSNR_{ROI}$ per frame for a fixed bit rates.

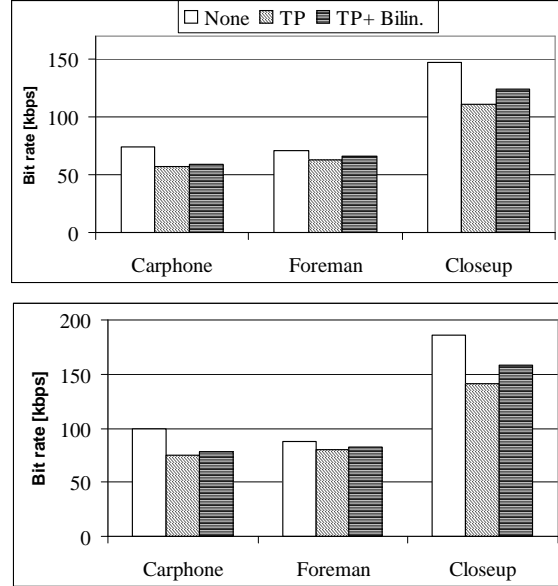


Figure 4.7: Bit rate (kbps) for $Qp = 28$ at frame rates of 10 fps (top) and 15 fps (bottom)

4.5.1 Bitrate

TP filtering alone (method 1) saves approximately 25% in bit rate during the encoding when using a fixed quantization parameter $Qp = 28$ for the Carphone and Closeup sequences (See figure 4.7). However, only 10% of the bits are saved for the Foreman sequence, which is mainly due to its low background motion content compared to that for the ROI. In figure 4.7 it can also be seen that TP filtering together with bilinear filtering (method 2) gives a reduction in bit rate of 21% for Carphone, 15% for Closeup and 6% for Foreman. Thus the TP filter including bilinear interpolation increases the bit rate by approximately 5% when compared to TP filtering on its own. The increase is larger for the Closeup sequence, which is partially caused by misdetections in the skin-color detection, where detected skin-color-like pixels in the background increases the size of the transition region. In addition the ROI is larger in the Closeup sequence compared to the others, which causes a larger part of the non-ROI area to be labeled as a transition region. On average the transition region costs more bits to encode, since all the blocks not skipped in the original sequence contain new information compared to the background which contains no new information in the even frames and therefore can be skipped in the encoding. Thus the

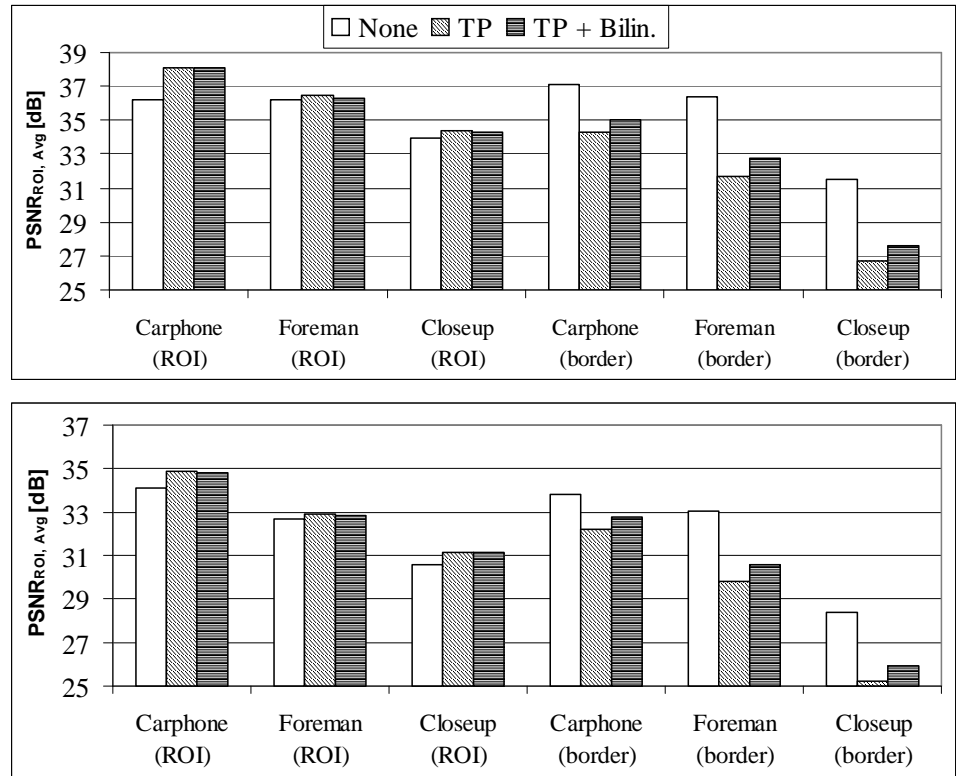


Figure 4.8: Average PSNR (dB) for 10 fps of the ROI at bit rates of 64 kbps (top) and 32 kbps (bottom).

more non-ROI pixels classified as transition region the less bits are saved.

4.5.2 PSNR of the ROI and the transition region

Tests were also performed with a fixed bit-rate by choosing the rate control option of the codec. At 64 kbps and 10 fps improvements within the ROI of 0.98 dB (Carphone) and 0.29 dB (Foreman) are obtained for temporal filtering without bilinear interpolation (See figure 4.8). In most cases it would be assumed that an improvement of over 1 dB is necessary to provide visible improvement. However, in this case, the details such as the eyes and the mouth are important to the viewer. Thus there is a visible improvement in the carphone sequence. A similar improvement as for 10 fps was achieved for 15 fps for all sequences. This is a moderate improvement in comparison to the reduction in bit rate presented in section 4.5.1, when the quantization parameter was fixed. The bits released during the encoding of the

background for the temporally filtered sequence when compared to the encoding of the original sequence are applied in order to reduce the error in each frame and are therefore applied to blocks with a high prediction error. In the spatial filter presented in chapter 3 the filter itself reduces the prediction error enabling reallocation of the bits to the high prediction errors, which are still present in the ROI after filtering. However, the temporal filter does not affect the prediction error and therefore the released resources are used in the complete image unless some additional control is added.

Figure 4.8 also shows that adding bilinear interpolation to the temporal filter improves the average border PSNR by 0.44-0.75 dB (Carphone) and 0.53-1.05 dB (Foreman). This is achieved by means of only a minor reduction of average PSNR for the ROI. An example of a frame with border effects which have decreased by using bilinear filtering can be viewed in figure 4.9, where it can be seen that the bilinear filter reduces boundary effects although not completely. However when the complete sequence is viewed the boundary effect in the bilinear case does not flicker as much as without bilinear interpolation.

4.6 Chapter summary

In this chapter two temporal filtering approaches were presented. The first removes background information from every second frame by replacing the background in the current frame with the background from the previous frame. To cope with the problem of artifacts at the ROI border appearing due to movements of position of the ROI, a second approach is presented, which extends the first temporal filter by including bilinear interpolation of the transition area from the ROI to the background. The results of the theoretical analysis and the quantitative tests are summarized in table 4.1.

The first method gives a reduction in bit rate by 10-25 % for a fixed quantization parameter or an increase in $PSNR_{ROI,Avg}$ of 0.29-0.98 dB for 64 kbps. The temporal filter is not able to re-allocate the bits released by the filter exclusively to the ROI which thus explains the moderate increase in $PSNR_{ROI,Avg}$.

The author's contributions to the chapter includes:

- Extending the approach in [25] to a preprocessing approach using filters and solving the problem of moving ROI borders by bilinear interpolation.
- Analysing the effect of the temporal filter on the calculation of rate and distortion, which is a part of the automatic determination of coding parameters to



Figure 4.9: Frame 156 from the 25 kbps Carphone sequence has been TP filtered without bilinear interpolation (top,left) and encoded using h.264 for 64 kbps (bottom,left). The same for TP with bilinear interpolation is presented in (top,right) and (bottom, right), respectively.

Coding efficiency of the background.	Less bits are allocated to motion vectors.
Re-allocation from background to ROI.	The bits released by the TP filter are re-allocated to both the background and ROI, since the the DCT components or prediction error is only marginally affected by the TP filter.
Computational complexity	Only the bilinear part of the TP filter has a noticeable computational complexity. The complexity of this part is $4N_T$ operations per frame, where N_T = number of bits in the transition region.
Bit rate	<p>TP filtering without bilinear interpolation saves approximately 25% in bit rate except for the foreman sequence, where only 10% is saved. This is due to the motion content of the sequence.</p> <p>Bilinear interpolation causes a minor loss in bit rate savings. A large ROI or a large number of misdetections might further reduce the saving of bit rate.</p>
PSNR	<p>A moderate improvement in $PSNR_{ROI, Avg}$ of 0.98 dB for the carphone sequence and 0.28 dB for the foreman sequence with the H.264 codec. The $PSNR_{ROI, Avg}$ remains approximately the same when bilinear interpolation is introduced.</p> <p>The $PSNR_{Border, Avg}$ is improved by 0.44-1.05 dB, when using bilinear interpolation compared to without bilinear interpolation.</p>

Table 4.1: A summary of the results of the theoretical analysis and the qualitative tests performed on the TP filter

achieve a target bit rate based on the general hybrid block-based encoder and standards.

- A computational complexity analysis.
- Test results from both objective measures such as PSNR and bit rate and an analysis of these.

Chapter 5

Spatio-temporal filtering

The improved coding efficiency of the background is mainly concentrated to the DCT coefficients and prediction error for the SP filter in chapter 3 and motion vectors for the TP filter in chapter 4. Thus by combining these two filters an increase in coding efficiency in the background is achieved and the reallocation problem of the TP filter is solved, since the SP filter reduces prediction error. The same ROI detection and determination of the quality map as addressed in section 3.2.2 is used to control both the SP and the TP part of the spatio-temporal filter (SPTP). The parts of the SPTP filter are described in figure 5.1. The SP filter in figure 5.2 is applied to every odd frame \mathbf{I}_{odd} as described in section 5.1. The even frames \mathbf{I}_{even} are TP filtered (See figure 5.3) using the previously spatially filtered odd frame $\mathbf{I}_{odd,SP}$ which is addressed in section 5.2. The performance of the SPTP filter is first evaluated by analysing the coding efficiency and ability to reallocate bits from the background to the ROI in section 5.3 and the computational complexity in 5.4. Thereafter test results are presented and analysed in section 5.6.

5.1 The SP filter

The SP filter in section 3.2.3 can be combined with the TP filter without any alterations, since only information within the same frame is used in the calculations.

5.2 The TP filter

Alterations of the TP filter in section 4.1 are necessary because the present spatio-temporal filter uses information from the SP filtered odd frames $\mathbf{I}_{odd,SP}$. The back-

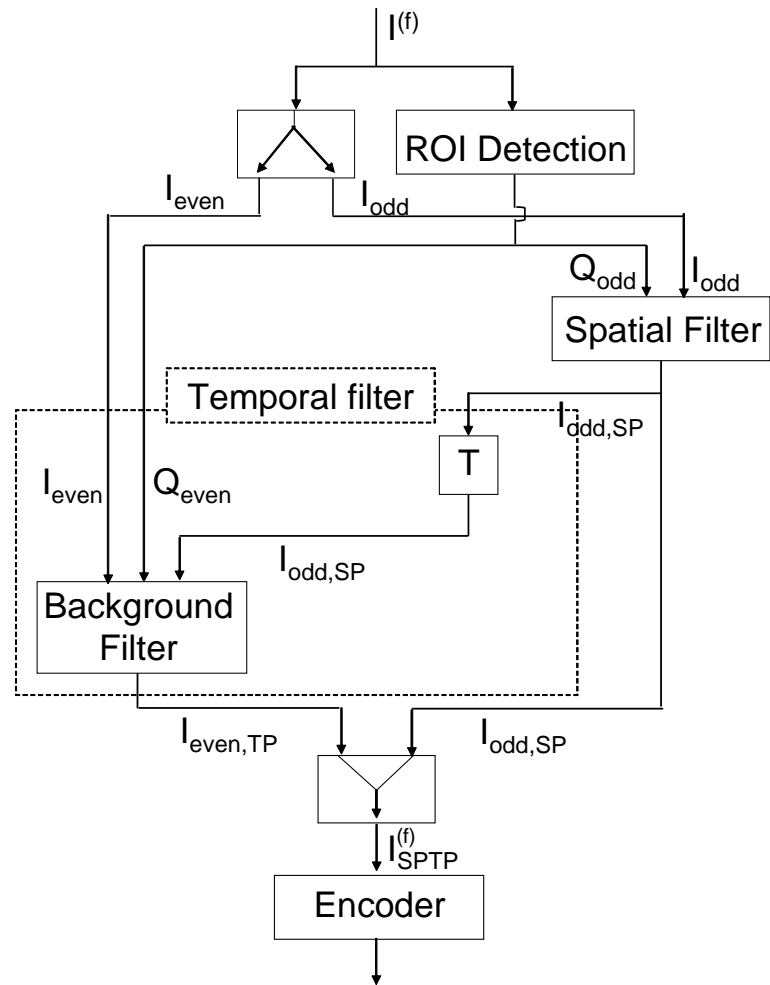


Figure 5.1: An overview of the SPTP filter, where the SP part is presented in detail in figure 5.2. and the TP part in figure 5.3.

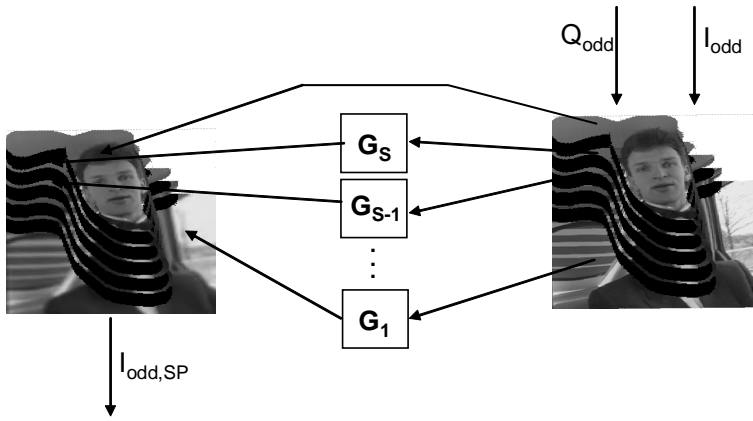


Figure 5.2: The SP part of the SPTP filter corresponds to the SP filter in section 3.2.3.

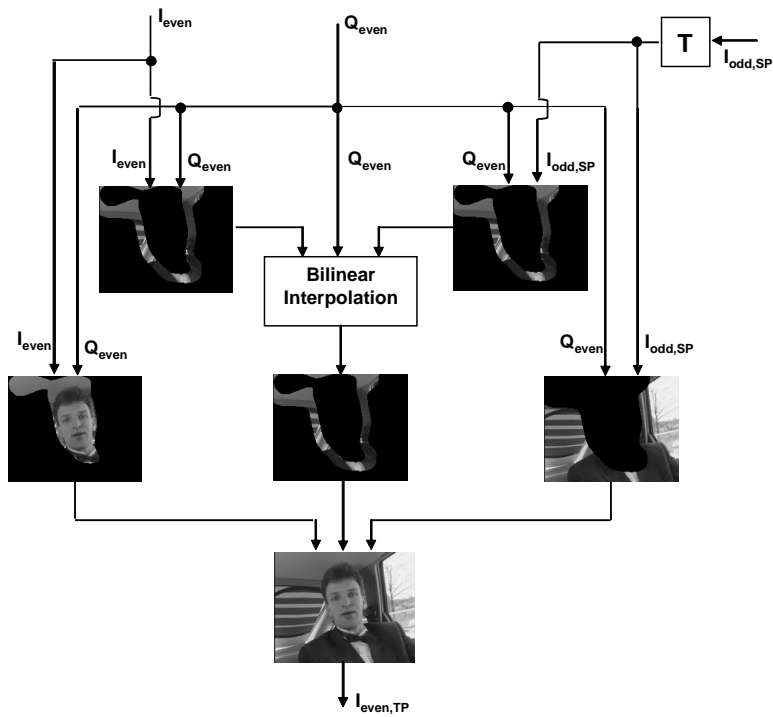


Figure 5.3: The TP part of the SPTP filter corresponds to the TP filter with bilinear interpolation in section 4.1 with a few modifications.

ground pixels of even frames used in the bilinear interpolation must also be SP filtered. Another technicality is that the determination of the ROI is block-based for the TP filter while pixel-based for the SP filter. Hence, there exist pixels classified as background by the SP filter that are simultaneously part of a block classified as ROI by the TP filter. Therefore, these pixels must be detected and SP filtered to ensure that the border stays as smooth as possible in all frames.

These alterations result in the following TP filter using values from the even frame I_{even} and the previous SP filtered odd frame $I_{odd,SP}$. First it is decided whether a block $B_{TP}(p, q)$ belongs to the ROI, the transition area or the background :

$$\mathbf{I}_{even,SPTP}^{(p,q)} = \begin{cases} f_{ROI}(\mathbf{I}_{even}^{(p,q)}, \mathbf{I}_{odd,SP}^{(p,q)}), & \text{if } Q^{(p,q)} \cap A_{ROI} \neq \emptyset \\ f_{bl} = (\mathbf{I}_{even}^{(p,q)}, \mathbf{I}_{odd,SP}^{(p,q)}), & \text{if } A_{Bg} \leq Q^{(p,q)} < A_{ROI} \\ \mathbf{I}_{odd,SP}^{(p,q)}, & \text{otherwise.} \end{cases}$$

Then for each pixel $(m, n) \in B_{TP}(p, q)$

$$f_{ROI}(\mathbf{I}_{even}^{(m,n)}, \mathbf{I}_{odd,SP}^{(m,n)}) = \begin{cases} f_{SP}(\mathbf{I}_{even}^{(m,n)}), & \text{if } (m, n) \cap ROI = \emptyset \\ \mathbf{I}_{even}^{(m,n)}, & \text{otherwise} \end{cases}$$

$$f_{bl}(\mathbf{I}_{even}^{(m,n)}, \mathbf{I}_{odd,SP}^{(m,n)}) = \alpha f_{SP}(\mathbf{I}_{even}^{(m,n)}) + (1 - \alpha) \mathbf{I}_{odd,SP}^{(m,n)}$$

$$\alpha = Q^{(m,n)} / A_{ROI}$$

The filter $f_{SP}(\mathbf{I}_{even}^{(m,n)})$ gives the SP filtered value for the pixel (m, n) .

5.3 Coding efficiency of the background and reallocation

The SPTP filter gives a larger reduction in bits allocated to the background than the SP and TP filters, since the SPT and TP parts of the SPTP filter reduces the number of bits in different parts of the encoded information. The analysis in section 3.3.1 indicates that the SP filtering of the even frames gives a reduction in the number of DCT components in the case of intra coding and decreases in the bits used to encode the prediction error in the case of inter coding. The TP filter, on the other hand, only decreases the amount of bits necessary to encode the motion vectors and the type of predictive coding (See section 4.2). The reduction is however not the sum of the reduction of the SP and TP parts, since the reduced prediction error has an impact on which motion vector is chosen as mentioned in section 3.3.2.

The bits released due to increased coding efficiency is used to improve the overall distortion and thus applied to where they reduce the distortion the most. In the

rate-distortion optimization of SP-filtered sequences (See section 3.3.2) these bits will mainly be reallocated to the ROI, since the distortion is reduced if more bits are used to encode a large prediction error. The SP filtering reduces the prediction error and therefore the bits are reallocated to the ROI, where the prediction error remains unaltered. In the TP case the bits are used to reduce distortion in the background as well as the ROI, since the prediction error remains approximately the same. Therefore, combining the SP and TP filter improves the reallocation of bits released by the TP filter to the ROI.

5.4 Computational complexity

The computational complexity is defined as the number of operations, where an operation consists of additions, subtractions, multiplications and divisions. The computational complexity of the detection and quality map is determined and presented in section 3.4.

The spatial filter including the reduction of computational complexity described in section 3.4.2 is used for odd frames, which gives computational complexity of $4L(N_{Bg} - N_{Bg, Skip}) + 4N_{Bg}$. Considering the even frames the result for the temporal filter in section 4.3 gives a complexity of $4N_{Tr}$. However, this computational complexity increases with $4L$ per pixel in the transition region N_{Tr} , since the pixel in the even frame $\mathbf{I}_{even}^{(m,n)}$ requires SP filtering. Thus, the computational complexity per I_{even} becomes $(4L + 4)N_{Tr}$. This gives an average computational complexity per frame of

$$2L(N_{Bg} - N_{Bg, Skip}) + 2N_{Bg} + 2(L + 1)N_{Tr}.$$

5.5 Experimental setup

The experimental setup for the ROI detection and quality map are specified in section 3.5. One additional sequence Closeup is added to the tests. This sequence is filmed by the author and contains a close-up of a talking face with a panning outdoor background. Therefore the background information of this sequence has an high level of motion. The ROI of closeup sequence is detected using the same method as the other sequences with a threshold of 32%. This gives an average ROI size of 49% of the frame.

The temporal filter with bilinear interpolation is applied for the temporal part of the spatio-temporal filter, since it was shown to give a better performance in chapter 4. A spatial filter, which consists of a set of 9 low-pass filters with variances

	Bit rate	Carphone	Foreman	Closeup
H.264	max	55 kbps	60 kbps	160 kbps
H.264	min	21 kbps	24 kbps	70 kbps
MPEG-2	max	160 kbps	220 kbps	360 kbps
MPEG-2	min	70 kbps	95 kbps	160 kbps

Table 5.1: The bitrates used in the tests where the max bit rate and min bit rates corresponds to 34 dB and 30 dB in $PSNR_{Avg}$, respectively.

controlled by the largest variance is $\sigma_1^2 = 5^2$ (See section 3.2.3), is applied on the odd frames. In addition the original sequence was encoded and used as a reference in the tests. The same codecs as in section 3.5 were applied in the tests using the bitrates defined in table 5.1.

The performance is measured using the objective quality measure, average PSNR of the ROI $PSNR_{ROI,Avg}$. In addition tests with a fixed quantization parameter of $Q_p = 28$ is performed, which shows how many bits are necessary to encode the sequence with this quantization parameter for the DCT components and the prediction error.

5.6 Experimental results

In this section the test results of the spatio-temporal filter are presented and analysed. First the bitrate of the encoder given a fixed quantization parameter giving an indication of the decrease in coding efficiency. This is followed by measuring the $PSNR_{ROI,avg}$ and $PSNR_{Bg,avg}$ for a target bitrate showing how the released bits are distributed within the video sequence.

5.6.1 Bitrate

The spatio-temporal filter saves about 31-52% of the bit rate compared to the original sequence (See figure 5.4), when the sequences were encoded with H.264 and 25 fps. The MPEG-2 encoded spatio-temporal filtered sequences show a decrease in bit rate of 22-36% instead. The bit rate decrease of the two different encoders can not be

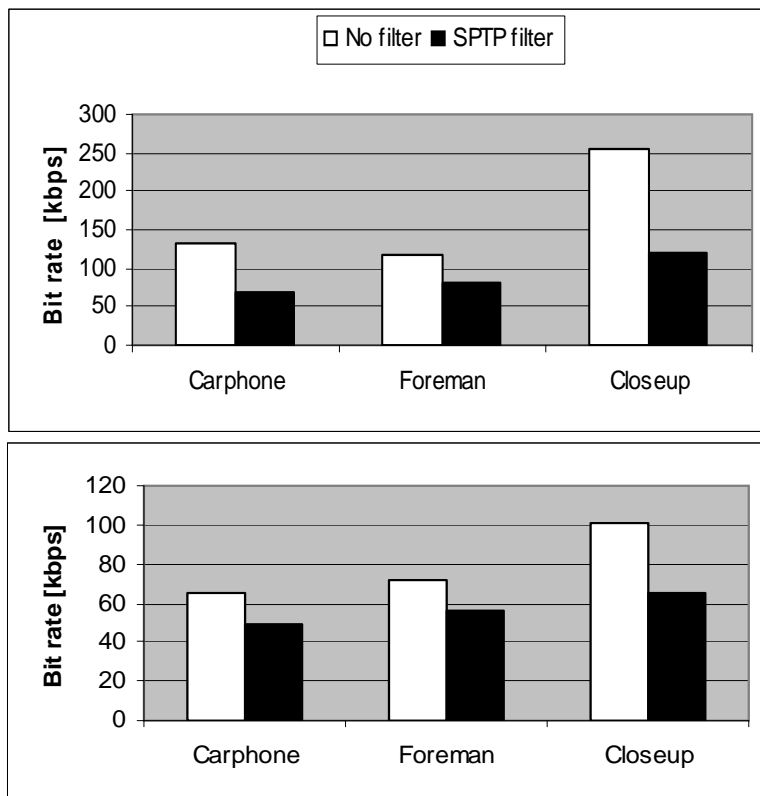


Figure 5.4: The bitrate in kbps for different video sequences filtered by no filters and with the spatio-temporal filter for H.264 (top) and MPEG-2 (bottom).

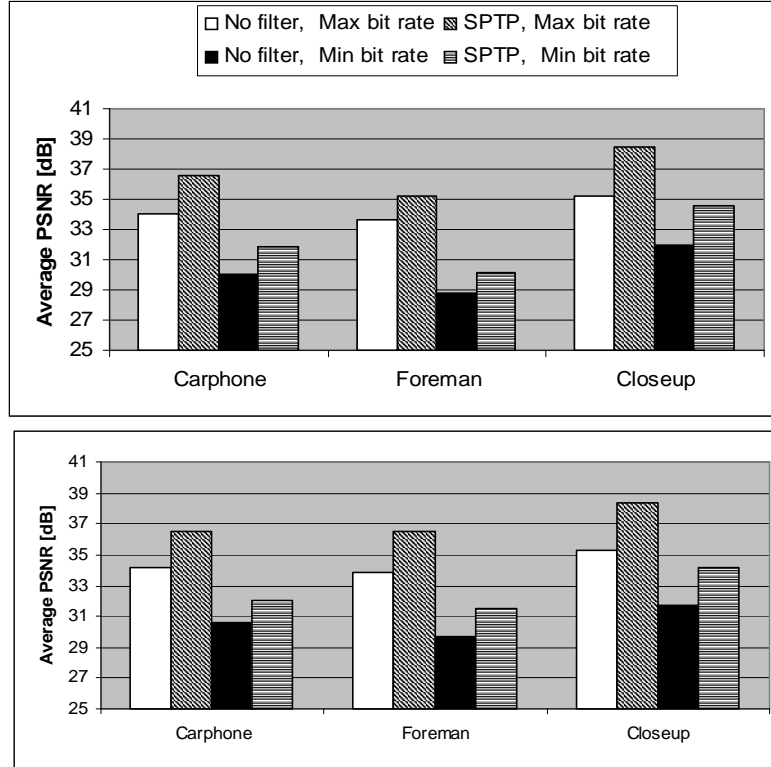


Figure 5.5: The average PSNR in dB for three different video sequences filtered by no filters and with the spatio-temporal filter at max and min bit rate using H.264 (top) and MPEG-2 (bottom).

directly compared since a fixed quantization parameter in one will not give a similar PSNR as the other. However it is clearly indicated that the spatio-temporal filter gives an clear decrease in bit rate.

5.6.2 PSNR

Tests were also performed with a fixed bit-rate by choosing the rate control option of the codec. In figure 5.5 it is shown that for the max target and min target bit rates encoding the spatio-temporal filtered sequences using H.264 gives an improvement in $PSNR_{ROI,Avg}$ of 1.58-3.2 dB and 1.32-2.04 dB, respectively, compared to encoding the original sequence.

Encoding with MPEG-2 instead results in an improvement in $PSNR_{ROI,Avg}$ of 2.38-3.07 dB (max bit rate) and 1.45-2.5 dB (min bit rate), respectively.

5.7 Chapter summary

In this chapter the spatial filter of chapter 3 and the temporal filter of chapter 4 were combined into one filter for three main reasons. First the spatial filter reduces the number of bits used for DCT components and prediction error, while the temporal filter reduces the number of bits used to encode motion vectors. Therefore by combining the two filters the coding efficiency of the background is increased. Secondly the problem that the bits released by the temporal filter is used to improve areas with a large prediction error in both the background and the ROI is solved since the spatial filter reduces this prediction error in the background. The third reason is to reduce the computational complexity imposed by the spatial filter. A summary of the results of the theoretical analysis and the experimental results can be found in table 5.2.

The authors contribution to the chapter includes:

- A combination of the spatial filter in chapter 3 and the temporal filter in chapter 4 into a spatio-temporal filter.
- An analysis of how a combination of the SP and the TP filter effect the coding efficiency of the background, the reallocation of bits from background to the ROI and the coding efficiency.
- Experimental results measuring PSNR and bitrate together with an analysis of these.

Coding efficiency of the background.	Less bits allocated to DCT coefficients, prediction error and motion vectors. The increase in coding efficiency of the motion vectors is both due to decreased prediction error (SP) and fewer motion vectors (TP).
Re-allocation from background to ROI.	The bits released by the SP filter and TP filter parts is mostly reallocated to the ROI where the most DCT components are present or the prediction error is the largest.
Computational complexity	Assuming $L \times L$ filter kernels in the SP part gives: $2L(N_{Bg} - N_{Bg, Skip}) + 2N_{Bg} + 2(L + 1)N_{Tr}$. N_{Bg} = number of pixels in the background $N_{Bg, Skip}$ = number of pixels skipped in the filtering. N_{Tr} = number of pixels in the transition region.
Bit rate	The SPTP filter saves 29-53% in bit rate when encoding with H.264 and 22-36% when encoding with MPEG-2.
$PSNR_{ROI, Avg}$	H264 codec: Increase with 1.58 - 3.2 dB for max bit rate and 1.32 - 2.04 dB for min bit rate. MPEG-2 codec: Increase with 2.38 - 3.07 dB for max bit rate and 1.45 - 2.5 dB for min bit rate.

Table 5.2: A summary of the results of the qualitative and quantitative analysis of the SPTP filter

Chapter 6

Qualitative and quantitative comparision of the filters

In this chapter the three filters proposed in this thesis, the SP filter in chapter 3, the TP filter in chapter 4 and the SPTP filter in chapter 5 are compared both analytically and experimentally. A summary of the qualitative analysis of the coding efficiency of the background and the reallocation of bits from the background to the ROI is presented in section 6.1. Thereafter follows a comparision of the computational complexities of the three filters in section 6.1.1. In addition a comparision of experimental results from the previous chapters together with additional tests on motion vector lengths and subjective tests are presented in section 6.3

6.1 Comparision of qualitative tests.

A summary of the qualitative tests in the previous chapters is presented in table 6.1. It can be seen that the coding efficiency is improved by using the SPTP filter instead of the SP and TP filter, since it removes bits from more parts of the encoding. However the decrease is not equal to the sum of the results for the SP and TP filters since both reduce the number of bits given to motion vectors in different ways. The re-allocation of bits is succesful for both the SP and SPTP filters, since both have a reduced prediction error in the background causing bits to be re-allocated from the background to the ROI.

Coding efficiency of the background.	<p>SP: Less bits allocated to DCT coefficients, prediction error and motion vectors, due to decreased prediction error.</p> <p>TP: Less bits allocated to motion vectors due to fewer motion vectors.</p> <p>SPTP: A combination of the two above.</p>
Re-allocation from background to ROI.	<p>SP: The release bits are mostly reallocated to the ROI where the majority of the DCT components are present or the prediction error is the largest.</p> <p>TP: The released bits are reallocated both to the ROI and the background.</p> <p>SPTP: As for the SP case.</p>
Computational complexity	<p>SP: Assuming $L \times L$ filter kernels in the SP part gives:</p> <p>SP: $4L(N_{Bg} - N_{Bg,skip}) + 4N_{Bg}$ operations per frame. N_{Bg} = the number pixels in the background $N_{Bg,skip}$ = the number of pixels skipped in the filtering</p> <p>TP: $4N_{Tr}$ operations per frame. N_{Tr} = number of pixels in the transition region of an odd frame.</p> <p>SPTP: $2L(N_{Bg} - N_{Bg,skip}) + 2N_{Bg} + 2(L + 1)N_{Tr}$.</p>

Table 6.1: A summary of the results of the qualitative analysis of the three filters presented in chapters 3 - 5.

6.1.1 Computational complexity

The computational complexity is defined as the number of operations, where an operation consists of additions, subtractions, multiplications and divisions. The computational complexity of the detection and quality map are determined and presented in section 3.4. This computational complexity is the same for all filters and therefore is disregarded in the calculations.

In table 6.1 the computation complexities of the three filters are presented. The computational complexity of the TP filter is much lower than for the SPTP filter or using SP filter by itself. However, it requires additional methods, such as low-pass filtering of the background, in order to re-allocate the released bits from the background to the ROI. Therefore, only the SP and the SPTP filters are considered in the comparison.

The SPTP filter gives a lower computational complexity than the SP filter if

$$2L(N_{Bg} - N_{Bg,Skip}) + 2N_{Bg} + 2(L + 1)N_{Tr} < 4L(N_{Bg} - N_{Bg,Skip}) + 4N_{Bg}$$

$$\implies 2(L + 1)N_{Tr} < 2L(N_{Bg} - N_{Bg,Skip}) + 2N_{Bg}$$

$$\implies N_{Tr} < N_{Bg} - \frac{L}{L + 1}N_{Bg,Skip}$$

$$\implies N_{Tr} < N_{Bg} - N_{Bg,Skip} < N_{Bg} - \frac{L}{L + 1}N_{Bg,Skip}$$

Thus the SPTP filter has a lower computational complexity than the SP filter if the number of SP filtered pixels ($N_{Bg} - N_{Bg,Skip}$) is larger than the number of pixels in the transition region in the even frames (N_{Tr}). In the case of no bilinear interpolation there is no transition area, $N_{Tr} = 0$. Thus without bilinear interpolation the SPTP filter always has a lower computational complexity than the SP filter.

6.2 Experimental setup

The same setup as for the SP filter in section 3.5, the TP filter in section 4.4 and the SPTP filter in section 5.5 is applied in the tests. However for the spatial filter only 9 filters and a maximum variance of $\sigma_1^2 = 5^2$ are applied and the version using bilinear interpolation is applied in tests concerning the TP filter. The max and min target bit rate is defined in section 5.5.

6.2.1 Motion vector analysis

In these tests 10 frames of each sequence were encoded using H.264 and the components of the transmitted motion vectors were extracted. From this data the average length of the motion vector x and y components $(d_{MV,1}^{(f,(k,l),e)}, d_{MV,2}^{(f,(k,l),e)}) = \bar{d}_{MV}^{(f,(k,l),e)}$ for non-zero motion vectors and the standard deviation were calculated using

$$m_{MV} = \frac{1}{N_{mv}} \sum_{f=1}^F \sum_{k=1}^{M/16} \sum_{l=1}^{N/16} \sum_{e=1}^{N_{mv}^{(f,(k,l))}} (|d_{MV,1}^{(f,(k,l),e)}| + |d_{MV,2}^{(f,(k,l),e)}|)$$

$$m_{C,MV} = \frac{1}{N_{C,mv}} \sum_{f=1}^F \sum_{(k,l) \in C} \sum_{e=1}^{N_{C,mv}^{(f,(k,l))}} (|d_{C,MV,1}^{(f,(k,l),e)}| + |d_{C,MV,2}^{(f,(k,l),e)}|)$$

$$\sigma_{MV}^2 = \frac{1}{N_{mv}} \sum_{f=1}^F \sum_{k=1}^{M/16} \sum_{l=1}^{N/16} \sum_{e=1}^{N_{mv}^{(f,(k,l))}} ((d_{MV,1}^{(f,(k,l),e)} - m_{MV})^2 + (d_{MV,2}^{(f,(k,l),e)} - m_{MV})^2)$$

$$\sigma_{C,MV}^2 = \frac{1}{N_{C,mv}} \sum_{f=1}^F \sum_{(k,l) \in C} \sum_{e=1}^{N_{C,mv}^{(f,(k,l))}} ((d_{C,MV,1}^{(f,(k,l),e)} - m_{C,MV})^2 + (d_{C,MV,2}^{(f,(k,l),e)} - m_{C,MV})^2)$$

where N_{mv} and $N_{C,mv}$ are the total number of motion vectors and the number of motion vectors within region $C \in (Bg, ROI)$ in the sequence. The number of motion vectors within the makroblock $MB(k, l)$ is given by $N_{mv}^{(f,(k,l))}$, since there can be more than one motion vector pair in H.264. The determination regarding whether a motion vector belongs to the ROI or background was made per MB such that for an MB with index $(k, l) \in ROI$ if $Q^{(m,n)} \geq A_{ROI}$ for $(m, n) \in MB(k, l)$.

6.2.2 Subjective tests

The setup stated at the beginning of this experimental setup section is applied in the tests. However, only the min target bit rate is applied in order to limit the number of tests.

The three main goals of this test in order of priority are:

1. Verify that encoding video filtered using the three filters gives a better quality than when the original sequence is encoded.
2. Determine which of the filtering methods produces the best perceptual quality.
3. Determine whether the perceptual quality is reduced by the artifacts resulting from the filtering of the background.

-3	Much worse	Worse
-2	Worse	
-1	Slightly worse	
0	The same	The same
1	Slightly better	Better
2	Better	
3	Much better	

Table 6.2: The ITU-R comparison scale and an the translation of this scale to the better, same and worse scale also used in the tests.

This information is extracted by a subjective test using a stimulus comparison method, where video clips were played in pairs. The reference sequence was the first to be which was followed by the test sequence. The viewer was then asked to judge the quality of the test sequence compared to the reference sequence using the ITU-R Comparison scale [60] in table 6.2. The test was performed using 15 non-expert test subjects, as recommended in ITU-R BT.500-11 [60]. They were asked to assess the quality of the video sequence using the given scale with no addition information concerning the encoding. During a preliminary run of the test it became apparent that the test subjects over-analysed the test results when they are only asked to assess the quality but provided with no additional instruction. The goal of the test was to obtain an indication regarding of how they would perceive the quality in everyday life. However, the test subjects appeared to focus more on the task of providing a good quality assessment than in giving their first impression. Therefore the instructions were modified as follows:

1. It was emphasized that they should assess the video sequences as if they were being seen in a real life situation. It was also clearly stated that they were not being asked to give a perfect quality evaluation, but rather that we were curious about the manner in which they perceived the quality as individuals.

2. The test subjects were asked to imagine that they were in a conversation with the characters in the video sequences over an application such as a mobile phone or pda.

Each test subject viewed 22 pairs of video sequences, where the first two were used as a warm up. The content of the 22 pairs was varied over 3 test groups such that all possible filter combinations requiring testing were included while the original sequence was varied. The order in which each sequence pair was played was randomized in order to minimize the effects of fatigue and learning. Fatigue occurs due to the strain of concentrating on several sequences in a row and learning occurs when the same original sequence is used several times. If a sequence, independent of the filtering, is shown several times the person becomes used to its appearance and begins to look around in the sequence for information that would not be noticed during its first occurrence. Some of the sequence pairs were shown twice to each viewer to increase the number of viewings and provide an indication of consistency in the quality assessment. In that case the order in which the two identical sequence pairs are shown are independent of each other. At the conclusion of the test some qualitative questions were asked in order to obtain an indication regarding what was perceived to be good quality. The tests were performed on 15 inch screens with a resolution of 1024×728 pixels.

The quantitative results were measured both by taking the vote mean m_{vote} of all the votes assessed according to the ITU-R comparison scale in table 6.2 and by the percentage that experienced the sequence to be better, the same or worse for each sequence pair. The quantitative questions were analysed to discover any similarities in the answers and were combined with views concerning the tests expressed by the test subjects during informal discussions after the test had been concluded. The mean difference between the vote for a sequence pair shown to one test subject, m_{Diff} and the corresponding standard deviation σ_{Diff} were used as a measure of the reliability of the tests.

6.3 Experimental results

In this section a comparison of performance is presented with the bit rate for a fixed quantization parameter and the $PSNR_{ROI, Avg}$ for a max and min target bit rate as performance measures. A quantitative analysis of the effect of the filters on the motion vectors is presented and compared to the $PSNR_{ROI, Avg}$ and the bit rate. In addition the results for the subjective test using human test subjects are also presented.

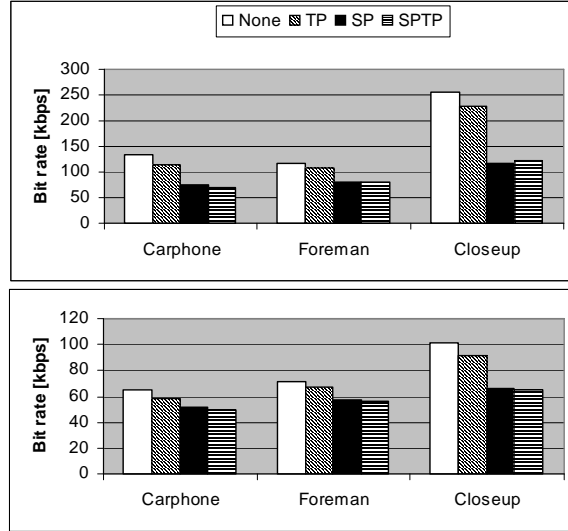


Figure 6.1: Bit rate (kbps) for $Qp = 28$ when encoding three different video sequences filtered using no filters or the three different filters at frame rates of 25 fps using H.264 (top) and MPEG-2 (bottom).

6.3.1 Bit rate

The largest reduction in bit rate compared to the original sequences (using h.264) was achieved by applying the SPTP filter (See the top of figure 6.1). The exception was the Closeup sequence for which the SP filter provides the largest reduction. It can also be seen at the top of figure 6.1 that the SP and SPTP filters give a reduction by at least a factor of two as that for the TP filter. Even though the bit rate for the SP and the SPTP filters differ by at most 5% the most the analysis in section 6.1.1 shows that the computational complexity of the SPTP filter is substantially lower. At the bottom of figure 6.1 the bit rates for the encoding using MPEG-2 are presented. MPEG-2 gives a propotional reduction in bit rate compared to the results for H.264. The SPTP filter gave the largest reduction for all sequences including the Closeup sequence.

6.3.2 PSNR

The SPTP filter gives the largest increase in $PSNR_{ROI,Avg}$ of the three filters compared to the unfiltered case (See figure 6.2 and figure 6.3), except for the closeup sequence. The SP filter gives almost the same increase in $PSNR_{ROI,Avg}$ as for the

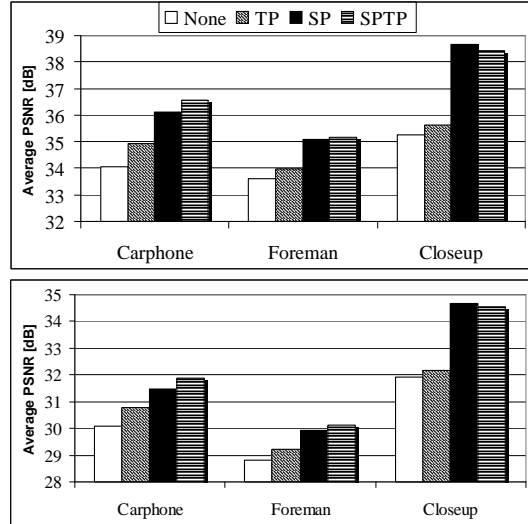


Figure 6.2: The $PSNR_{ROI, Avg}$ for the max bit rate (top) and min bit rate (bottom), when encoding using H.264 at frame rates of 25 fps to encode none, SP, TP and SPTP filtered

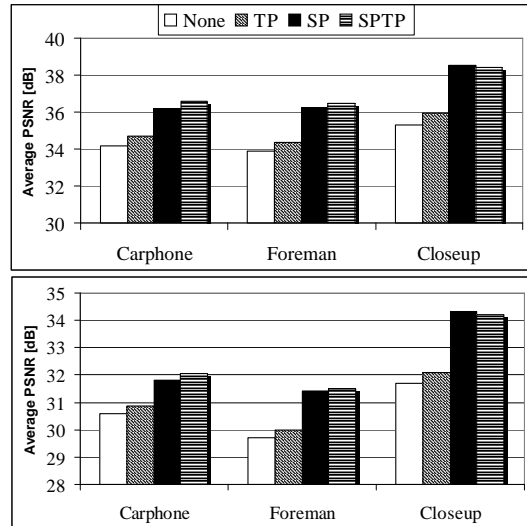


Figure 6.3: The $PSNR_{ROI, Avg}$ for the max bit rate (top) and min bit rate (bottom), when encoding using MPEG-2 at frame rates of 25 fps to encode none, SP, TP and SPTP filtered sequences.

SPTP filter and an even larger $PSNR_{ROI, Avg}$ for the Closeup sequence. The TP filter gives only a moderate improvement.

6.3.3 Motion vector analysis

In this section the motion vectors are extracted and analysed for the encoding using h.264 with a target bit rate as well as a fixed quantization parameter. There is a practical difference involved when encoding motion vectors with a target bit rate instead of a fixed quantization parameter which must be considered in this analysis (See section 3.3.2). In the case of fixed quantization parameters there is no limitation of bit rate and thus the motion vectors are chosen such that the sequence receives the minimum distortion under a particular quantization of the prediction error. However at a target bit rate only a lesser amount of motion vectors is afforded unless the target bit rate is high enough to include the rest. The release of bits by the filters therefore enables the use of more motion vectors to decrease the overall distortion. However, for the tested sequences the filtering reduces the number of motion vectors in a majority of the cases, when a fixed quantization parameter or a target bit rate is used (See figures 6.4 and 6.5).

The standard deviation was in the majority of the cases was much larger than the mean value and the change in standard deviation corresponded to the change in mean value. Thus the standard deviation was not analysed any further.

SP filter

In the case of the SP filter, the number of background motion vectors increases for the carphone and foreman sequences (See figure 6.5), when encoded at a target bit rate for low frame rates. A possible reason for this is that the prediction errors cost less to code due to the reduction in detail of the background. This might cause some of the saved bits to be used to reduce the distortion by adding more motion vectors. On the other hand, in the case of a fixed quantization parameter the SP filter reduces the number of motion vectors (See figure 6.4). This confirms the statement in 3.3.2 that the reduction of details could affect the choice of motion vectors. If the prediction error for no motion vector, $\bar{d}_{MV} = (0, 0)$, is reduced it may become cheaper to encode just this prediction error than the motion vector and prediction error chosen in the original sequence. However, this reduction in motion vectors could be partially the result of a reduced number of motion vectors within a makroblock.

Another observation is that the number of motion vectors in the ROI remains almost stationary for a fixed quantization parameter, the exception being the Closeup sequence. On the other hand, the number of motion vectors in the ROI increases in

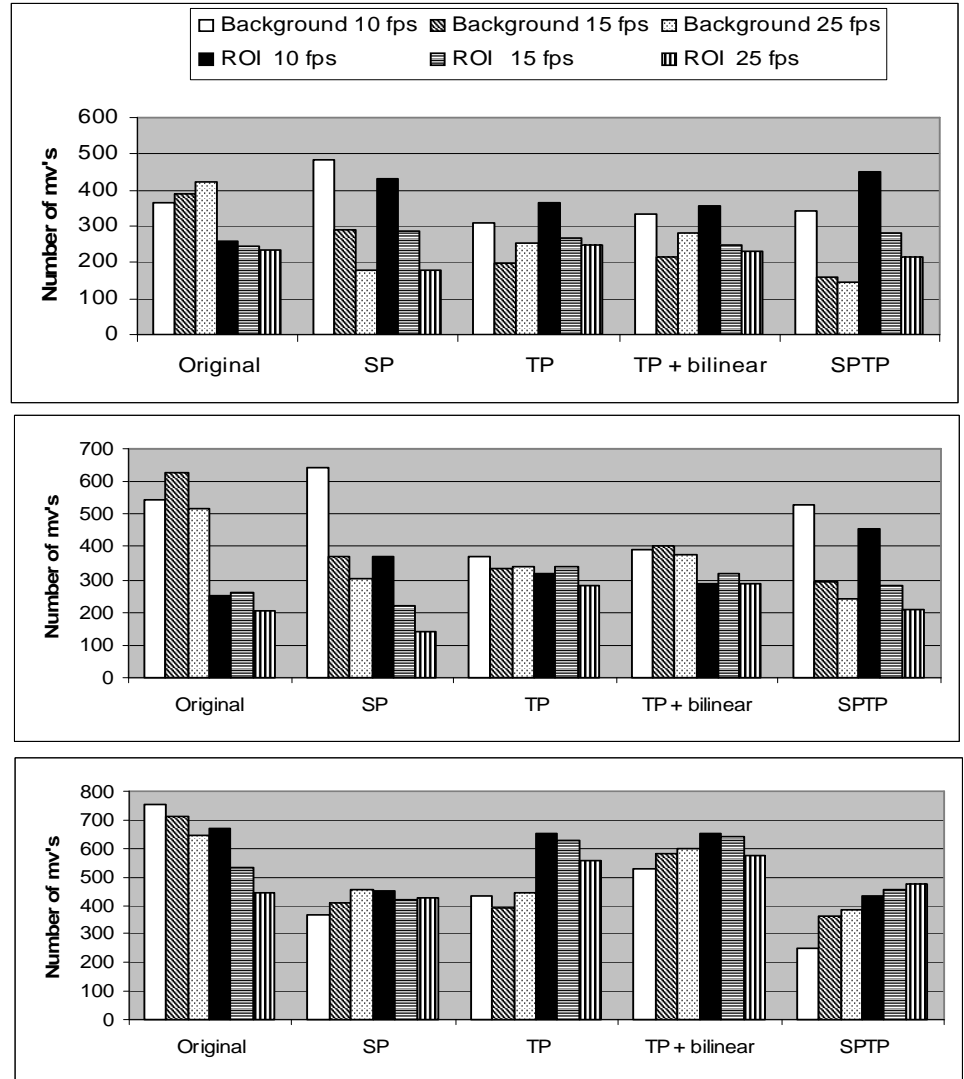


Figure 6.4: The number of motion vectors assigned to the background and the ROI for the carphone (top), forman (middle) and closeup (bottom), when the sequences are filtered using the different filters and then encoded using H.264 with fixed quantization parameter $Q_p = 28$.

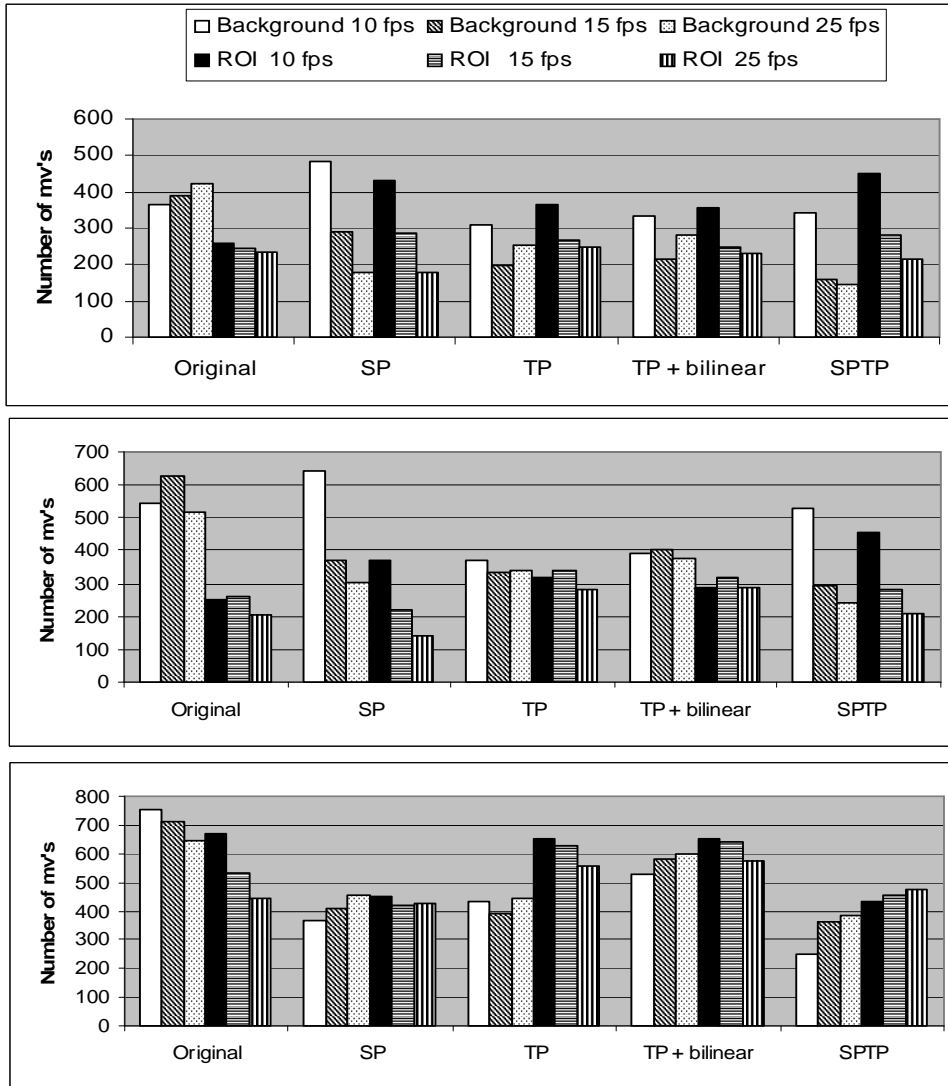


Figure 6.5: The number of motion vectors assigned to the background and the ROI for the carphone (top), forman (middle) and closeup (bottom), when the sequences are filtered using the different filters and then encoded using H.264 at max bit rate as target bit rate.

the majority of the cases when a target bit rate is applied. This confirms that the bits released from the background are used both to improve the encoding of the prediction error as well as the motion vectors of the ROI.

TP filter

There is a reduction in the number of background motion vectors by almost 50% for the TP filtered sequence compared to the original sequence in the case of a fixed quantization parameter in figure 6.4. This corresponds to the theoretical analysis in section 4.2.2. However, this reduction in the number of motion vectors is sometimes less when a target bit rate is applied. As for the SP filter this is most likely the case because adding motion vectors to the background of odd frames might reduce the overall distortion compared to encoding the original sequence. The addition of bilinear interpolation increases the number of motion vectors slightly for the carphone and the foreman sequences compared to the TP filtered sequences (See figure 6.5 and figure 6.4). This is to be expected as the blocks in the transition area contain information for the encoding of the difference in the background of the even frames. The larger increase in motion vectors in the closeup sequences occurs because the transition area covers more of the non-ROI than the other sequences.

The ratio of the number of motion vectors in the background and the number of motion vectors in the ROI affects the adaptive coding in H.264. The region containing the majority of the motion vector content has the greatest impact on the adaptation of codeword lengths to motion vectors statistics by CABAC. In figure 6.5 it can be seen that the TP filter provides a reduction in the number of background motion vectors for the closeup sequence. At the same time the number of ROI motion vectors in most cases increases or remains the same. Thus the adaptation will be in favor of the motion vectors from the ROI for the TP and SPTP filters. This provides an explanation for the lack of improvement in quality when the TP filter or the SPTP filter is applied.

In addition the mean lengths of the motion vector components were also extracted for each sequence and presented in figure 6.6. In the worst case scenario given in section 4.2.2 the non-zero motion vectors in the background of the TP filtered sequence are twice as long as those when the original sequence is encoded. The mean values of the TP filtered sequences encoded using a fixed quantization parameter show that the average motion vectors are substantially shorter than twice the mean when using no filter. The sequences showing the largest increase in mean value contains a large amount of background movement. This indicates that the assumption that a large variance in motion in the background in the original sequence causes a larger increase in motion vector lengths for the TP filtered sequence. This

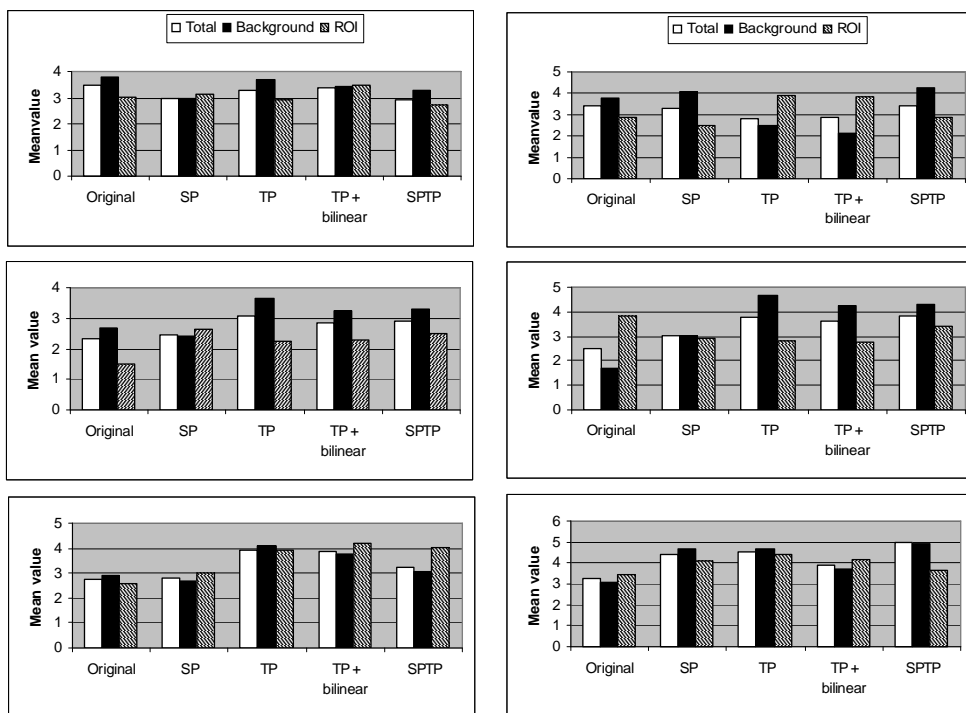


Figure 6.6: The mean length of the motion vectors assigned to sequences Carphone (top), Forman (middle) and Closeup (bottom), when the sequences are filtered using the different filters and then encoded using H.264 at 25 fps with a fixed quantization parameter (left) and a target bit rate (right).

is also the case in which a target bit rate is applied, apart from for the Foreman sequence.

SPTP filter

In figures 6.4 and 6.5 it can be seen that the reduction in the number of motion vectors of the SPTP filter compared to the case when using no filters is almost a weighted sum of the decrease of the TP and SP filters by themselves.

Summary

Based on the decrease in the number of motion vectors and a limited increase in motion vector lengths it can be concluded that the number of bits assigned to motion vectors in the background is substantially reduced, when the filters are applied. Although this decrease is reduced if the ROI contains the main part of the motion vectors. The codeword lengths and thus the number of bits to encode the motion vectors are then adapted to the motion vector statistics of the ROI. In addition the total number of bits assigned to inter-coded frames also includes the prediction error, which accounts for those cases when this decrease does not correspond to the increase in $PSNR_{ROI,Avg}$.

6.3.4 Subjective tests

The results of the subjective tests are investigated using two measures. The mean vote m_{vote} in figure 6.7 (According to the ITU-R comparison scale in table 6.2.) and the percentage of votes for better, same and worse are given in figures 6.8 and 6.9. The SP filtered and H.264 encoded closeup sequence were omitted from the results and the analysis. It can be seen that the effect of the TP filter is not clearly visible to most test subjects. This confirms the conclusion in the qualitative analysis that the bit re-allocation from the background to the ROI is not successful (See table 6.1). This is an explanation regarding the limited improvement in $PSNR_{ROI,Avg}$ of the TP filter presented in figure 6.2. The SP and particularly the SPTP filters show an improvement for the Carphone and Foreman sequence. In the comparison of the SPTP filter with respect to the other filters at the bottom of figure 6.7 it can be seen that the SPTP filter shows an improvement compared to the other filters for the Carphone and Foreman sequence. This corresponds well with the previous qualitative and quantitative tests.

The Closeup sequence was assessed to have no change in perceived quality when the filters were applied except for the TP and SPTP filter and 10 fps H.264 encoded-

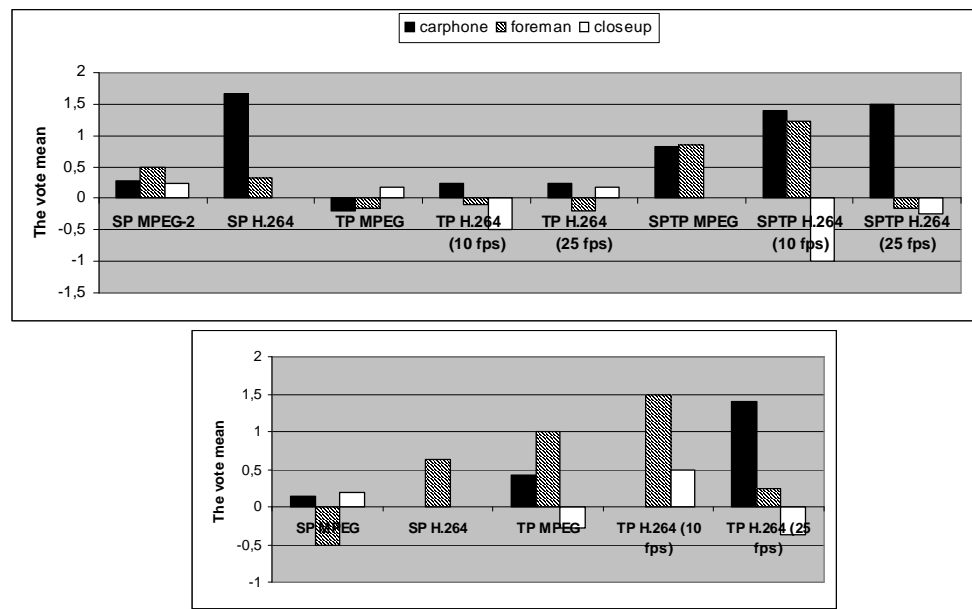


Figure 6.7: The vote mean of the different filters with no filter as a reference (top) and the vote mean of the SPTP filtered test sequence compared to the SP or TP filtered sequences (bottom).

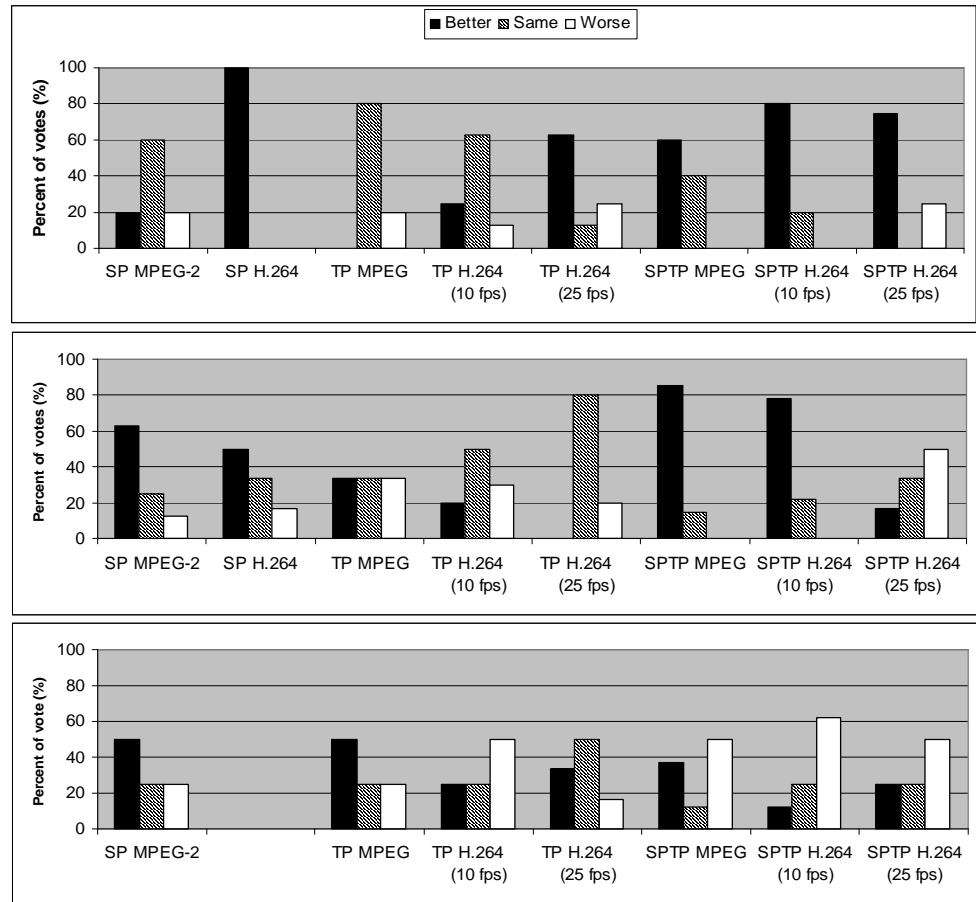


Figure 6.8: The percent of votes classified as better, same or worse according to the scale in table 6.2 with no filter as reference for Carphone (top), Foreman (middle) and Closeup (bottom).

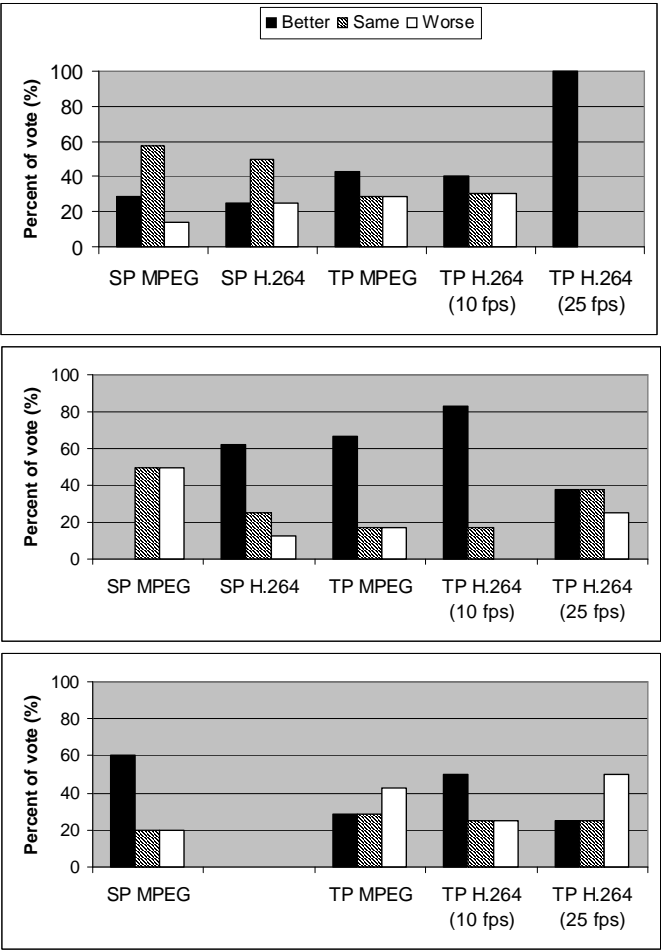


Figure 6.9: The percent of votes classified as better, same or worse according to the scale in table 6.2 for the SPTP filtered sequence compared to SP and TP filtered sequences for Carphone (top), Foreman (middle) and Closeup (bottom).

ing. The large movement in the background due to larger camera movement and moderate movement of the face results in a higher quality for the face as compared to the background. Thus the quality of the main features of the face are asufficiently high so that the movement of the lips can be followed in the majority of the cases. Therefore it is to be expected that an increase in $PSNR_{ROI, Avg}$ will not affect the overall perceived quality as much as in the other test sequences. In addition the movement in the background was experienced as being somewhat disturbing by the test subjects. The TP filter causes some additional jerkiness at high movement from frame to frame, which gives a possible explanation for the low bit rates performing rather worse than the high bit rates for the TP and SPTP filters. Further tests would be necessary to determine whether jerky movement in the background would attract attention even if the the perceive quality of the ROI was substantially improved. If that is the case a threshold for the maximum movement between frames or a post-processing step would reduce the artifacts such that they do not draw attention to the face. Another possible reason for the low results for the Closeup sequence is that the ROI border is located at a position rather close to the face.

The answers fpr the qualitative questions together with the discussions with the test subjects confirm that distortion to the face was experienced as the most disturbing. In particular that involving either the mouth or the eyes. In addition they found the quality in all sequences to be extremely bad, which is to be expected since they only viewed it at min bit rate. However this also made it more difficult for any improvements to be noticed due to the high number of strong artifacts attracting attention. In addition some people noticed that they tended to look for the problems in the image rather than focusing on the over all experience of quality when viewing the same type of sequence several times.

The reliability measures for the mean and standard deviation of the difference between results for identical tests were determined as $m_{Diff} = 1,11$ and $\sigma_{Diff} = 0.34$. Considering that the vote mean for most sequence pairs is $|m_{Vote, Pair}| \leq 1$ this verifies the statement by the test subjects that they experienced problems when assessing the quality. More reliable results could probably be achieved if a bit rate between the two extremes min and max bit rate, was used instead of only the min bit rate. In addition by including more sequences the effects of fatigue and learning are reduced. It might also be possible to test whether playing both sequences at the same time next to each other would be of assistance when judging them. However, it is difficult to view two things at once. The problem involved with playing them after each other is that the viewer is unable to remember the quality in the complete reference sequence when assessing the quality of the test sequences.

Filter	SP filter	Compared to TP filter	SPTP filter
SP	-	> 25%	-8% → 3%
TP	< -34%	-	< -34%
SPTP	-3% → 7%	> 25%	-

Table 6.3: The a summary of the percentage reduction in bits achieved by using one of the three filters compared to the others.

6.4 Chapter summary

According to the qualitative analysis summarized in table 6.1 the SPTP performs better than the other filters. It has the highest coding efficiency of the three filters and a lower computational complexity than the SP filter. This is based on the assumption that the transition area contains less pixels in a frame than the number of filtered pixels. The smaller the background area the less th number of bits to be found for re-allocation to the ROI. Therefore the background must be sufficiently large in order for it to be worthwhile applying ROI video coding. Thus the assumption stated above is valid. In addition the SPTP filter successfully re-allocates released bits from the background to the ROI. The motion vector analysis confirms the results of the qualitative analysis concerning the bit assignment to motion vectors.

The SPTP filter also has a better performance in the majority of the cases, when using the measures $PSNR_{ROI,Avg}$ and bit rate. (See table 6.4 and 6.3 .) However the SP filter gave a better result when encoding the Closeup sequence, which explains the negative minium value for both the bit rate and the $PSNR_{ROI,Avg}$ when the SPTP filter is compared to the SP filter. This is probably due to the larger number of motion vectors within the ROI than the background and thus has the most impact on codeword lengths. In addition the transition area of this sequence occupies substantially more of the background than in the other sequences.

The subjective test also indicated that the SPTP filter gives a better performance concerning perceived quality. The improvement in $PSNR_{ROI,Avg}$ and bit rate was not visible for the Closeup sequence for any of the filters. The extreme movement

Filter	Bit rate	SP filter	Compared to TP filter	SPTP filter
SP	max	-	> 1.11 dB	-0.44 dB → 0.25 dB
	min	-	> 0.71 dB	-0.38 dB → 0.09 dB
TP	max	< -1.11 dB	-	< -1.21 dB
	min	< -0.72 dB	-	< -0.92 dB
SPTP	max	-0.25 dB → 0.44 dB	> 1.21 dB	-
	min	-0.09 dB → 0.38 dB	> 0.92 dB	-

Table 6.4: The increase in $PNSR_{ROI, Avg}$ achieved by using one of the filters compared to the others.

of the background in this sequence could be a distraction for the viewer, and in particular when jerky movement is introduced by the TP filter. An additional aspect to consider was that the quality within the face was already sufficiently large in the facial region to reduce the impact of any improvement. However the reliability of the subjective tests is questionable and further tests are necessary so as to provide a conclusive analysis.

The authors contributions to the chapter includes:

- A summary of the analysis regarding how the encoding of a filtered sequence affects the coding efficiency of the background and the re-allocation of the released bits from the background to the ROI.
- A comparison of the computational complexities of the three different filters.
- An qualitative analysis of the effect encoding a filtered video sequence has on motion vector assignment.
- The results and analysis of a subjective test.

Chapter 7

Conclusions

7.1 Summary and discussion

ROI video coding makes it possible to adapt the encoding with regards to how a human would perceive the quality of a video sequence in a particular application at low bit rates. The quality of the ROI can be improved by reducing the quality in the less noticeable background, which gives the appearance of improved perceived quality to the viewer without having to increase the bit rate. In this thesis the focus was on how to re-allocate bits from the background to the ROI without altering the encoder, assuming that the ROI is correctly detected. This enables any arbitrary block-based encoder to be used to encode the video at the expense of adaptivity to changes in bit rate. Pre-processing in the form of filters removing information in the background can be used to achieve a successful re-allocation of bits from the background to the ROI. Three pre-processing methods were proposed. The spatial (SP) filter removing details in the background, the temporal (TP) filter which removes information in the background of every second frame and a combination of the two filters in the form an spatio-temporal (SPTP) filter.

The SP filter reduces the number of bits which are allocated to the DCT components, the prediction error and the motion vectors of the background due to the reduced prediction error. The use of multiple Gaussian filters to enable a gradual quality transition from ROI to background decreases the likelihood of creating artifacts at the ROI border due to a low cut-off frequency. In addition to reducing perceived quality at the border these artifacts also contain high frequencies. Thus the quality of the ROI remains the same or better when the gradual quality transition is used instead of just one low-pass filter. The TP filter, on the other hand, reduces the number of bits used for the motion vectors by decreasing the number of motion

vectors. Bilinear interpolation can be applied to reduce the artifacts that appears in the TP filtering due to large movements of the ROI border.

In order for the pre-filtering to result in a successful ROI video coding the bits released by the filter must be re-allocated to the ROI by the encoder. The encoder assigns bits to where they have the greatest effect on the reduction in the distortion. In intra frames that consists of the number of DCT components and in inter frames the prediction error. Therefore, the SP filter manages such a reallocation successfully whereas for the TP filter the bits are reallocated to the background as well as the ROI.

The SPTP filter combines characteristics of the SP and TP filters giving a reduction in the number of the bits allocated to the DCT components, the prediction error and the motion vectors. The number of bits assigned to motion vectors is reduced compared to either the TP or the SP filter, since the SP and TP filters affect the motion vector assignment in different ways. The re-allocation problem of the TP filter is also solved as the SPTP filter applies the TP part of the filter on SP filtered data. In addition it is also shown that the SPTP filter has a lower computational complexity than the SP filter as long as the number of pixels that are filtered in the odd frames is larger than the number of pixels in the transition area. This can be assumed to be true for natural video sequences and a maximum allowable ROI size. The concept of ROI video coding only works if the ROI size is limited, since there has to be a sufficient amount of bits in the background to be re-allocated. In addition the computational complexity of the SPTP filter is always lower than for the SP filter, when no bilinear interpolation is used. A summary of the qualitative analysis of the filters is found in table 7.1.

In tables 7.2 and 7.3 it is confirmed that the SPTP filter gives the best performance measured in bit rate and $PSNR_{ROI, Avg}$ in the majority of the cases. The negative results compared to the SP filter in both bit rate and $PSNR_{ROI, Avg}$ is produced during tests on the Closeup sequence. However the improvement in using the SP filter instead of the SPTP filter in this case is marginal. Therefore the lower computational complexity of the SPTP filter makes it the better choice.

The subjective tests gives a confirmation that the SPTP filter does improve perceived quality even if it is largely marginal. In addition there were indications that in the presence of extreme movement the TP filter might cause jerky artifacts that are strong enough to contribute to a reduced perceived quality at low frame rates. This requires further investigation to determine whether preventative measures such as a threshold on movement or post-processing are necessary. The overall reliability of the tests could be improved by additional tests with a higher bit rate, more sequences and perhaps a different test setup.

Coding efficiency of the background.	<p>SP: Less bits allocated to DCT coefficients, prediction error and motion vectors, due to decreased prediction error.</p> <p>TP: Less bits allocated to motion vectors due to fewer motion vectors.</p> <p>SPTP: A combination of the two above.</p>
Re-allocation from background to ROI.	<p>SP: The release bits are mostly reallocated to the ROI where the majority of the DCT components are present or the prediction error is the largest.</p> <p>TP: The released bits are reallocated both to the ROI and the background.</p> <p>SPTP: As for the SP case.</p>
Computational complexity	<p>SP: Assuming $L \times L$ filter kernels in the SP part gives:</p> <p>SP: $4L(N_{Bg} - N_{Bg,skip}) + 4N_{Bg}$ operations per frame. N_{Bg} = the number pixels in the background $N_{Bg,skip}$ = the number of pixels skipped in the filtering</p> <p>TP: $4N_{Tr}$ operations per frame. N_{Tr} = number of pixels in the transition region of an odd frame.</p> <p>SPTP: $2L(N_{Bg} - N_{Bg,skip}) + 2N_{Bg} + 2(L + 1)N_{Tr}$.</p> <p>The TP filter has the lowest computational complexity but it fails to reallocate most bits to the ROI. The SPTP filter on the other hand, has a lower computational complexity than the SP filter if $N_{Tr} < N_{Bg} - N_{Bg,skip}$.</p>

Table 7.1: A summary of the results of the qualitative analysis of the three filters.

Filter	None	SP filter	Compared to TP filter	SPTP filter
None	-	< -8%	< -45%	< -45%
SP	> 31%	-	> 25%	-8% → 3%
TP	> 8%	< -34%	-	< -34%
SPTP	> 31%	-3% → 7%	> 25%	-

Table 7.2: The a summary of the decrease in bits in percent achieved by using one of the three filters or no filters compared to the others.

7.2 Future works

Future works could include to:

- Perform additional subjective tests with other bit rates and additional sequences to obtain more reliable results and to examine when jerky movement in the background has an substantial effect on perceived quality.
- Improve the methods presented in section 2.1 for detecting the ROI. Even though extensive work is performed on face detection there is still a need for faster and more accurate methods. In addition other applications could be considered (See section 1.3.1), where methods of detection have not been thoroughly researched at the present time.
- Applying ROI video coding strategies to scalable video coding.
- Incorporate low pass filtering into the codec and adapt the rate-distortion optimization to the low pass filtering in addition to other methods and ordinary encoding.
- Improve the parts of the methods where temporal filtering is included to minimize the problem associated with a moving ROI even further. Include an optional post-processing step to reduce the impact of jerky movements in the

Filter Bit rate	None	Compared to SP filter	TP filter	SPTP filter
None				
max	-	< -0.37 dB	< -1.48 dB	< -1.58 dB
min	-	< -0.23 dB	< -1.12 dB	< -1.32 dB
SP				
max	> 1.48 dB	-	> 1.11 dB	-0.44 dB → 0.25 dB
min	> 1.12 dB	-	> 0.71 dB	-0.38 dB → 0.09 dB
TP				
max	> 0.37 dB	< -1.11 dB	-	< -1.21 dB
min	> 0.23 dB	< -0.72 dB	-	< -0.92 dB
SPTP				
max	> 1.58 dB	-0.25 dB → 0.44 dB	> 1.21 dB	-
min	> 1.32 dB	-0.09 dB → 0.38 dB	> 0.92 dB	-

Table 7.3: The increase in $PNSR_{ROI,Avg}$ achieved by using one of the filters or no filters compared to the others.

background. This post-processing would attempt to recreate the information in the background that was removed by the temporal filter.

Bibliography

- [1] Moving Pictures Experts Group. <http://www.chiariglione.org/mpeg/>.
- [2] International Telecommunications Union. <http://www.itu.int/home/>.
- [3] ISO/IEC 13818-2. Information technology - generic coding of moving pictures and associated audio information: Video. International Organization for Standardization, 1996.
- [4] ITU-T Rec. H.261. Video codec for audiovisual services at px64 kbit/s. International Telecommunications Union, 1993.
- [5] ITU-T Rec. H.263. Video coding for low bitrate communication. International Telecommunication Union, 1998.
- [6] ISO/IEC 14496-2. Information technology - coding of audio-visual objects part 2: Visual. International Organization for Standardization, 1999.
- [7] ITU-T Rec. H.264. Advanced video coding for generic audiovisual services. International Telecommunication Union, 2003.
- [8] Thomas Wiegand, Gary J. Sullivan, Gisle Bjontegaard, and Ajay Luthra. Overview of the H.264/AVC video coding standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 13:560–576, July 2003.
- [9] Yao Wang, Jörn Ostermann, and Ya-Qin Zhang. *Video Processing and Communications*. Prentice-Hall Inc., Upper Saddle River, New Jersey, USA, 2002.
- [10] Alexandros Eleftheriadis and Arnaud Jacquin. Automatic face location detection and tracking for model-assisted coding of video teleconferencing sequences at low bit-rates. *Signal Processing: Image Communication*, 7:231–248, September 1995.
- [11] Mei-Juan Chen, Ming-Chieh Chi, Ching-Ting Hsu, and Jeng-Wei Chen. ROI Video Coding Based on H.263+ with Robust Skin-Color Detection Technique. *IEEE Transactions on Consumer Electronics*, 49:724–730, August 2003.

- [12] Ariel Tankus and Yehezkel Yeshurun. Detection of Regions of Interest and Camouflage Breaking by Direct Convexity Estimation. In *1998 IEEE Workshop on Visual Surveillance*, pages 42–48, 1998.
- [13] Linda Karlsson. Detection of interesting areas in images by using convexity and rotational symmetries. Master Thesis No. 31 (2002), Dept. of Science and Technology, Linköping University, Sweden, 2002.
- [14] Jerome Meessen, Christophe Parisot, Xavier Desurmont, and Jean-Francois Delaigle. Scene Analysis for Reducing Motion JPEG 2000 Video Surveillance Delivery Bandwidth and Complexity. In *2005 IEEE International Conference on Image Processing, Genua, Italy*, volume 1, pages 577–580, 2005.
- [15] Guangyu Wang, Tien-Tsin Wong, and Pheng-Ann Heng. Real-Time Surveillance Video Display with Saliency. In *The Third ACM International Workshop on Video Surveillance & Sensor Networks*, pages 37–43, 2005.
- [16] John D. McCarthy, M. Angela Sasse, and Dimitros Miras. Sharp or Smooth? Comparing the effects of quantization vs. frame rate for streamed video. In *The SIGCHI Conference on Human Factors in Computing Systems*, pages 535–542, 2004.
- [17] Yu-Lin Kang, Joo-Hwee Lim, Qi Tian, Mohan S. Kankanhalli, and Chang-Sheng Xu. Visual Keywords Labeling in Soccer Video. In *The 17th International Conference on Pattern Recognition*, pages 535–542, 2004.
- [18] Laurent Itti. Automatic Foveation for Video Compression Using a Neurobiological Model of Visual Attention. *IEEE Transactions on Image Processing*, 13:1304–1318, October 2004.
- [19] C. Koch and S. Ullman. Shifts in selective visual attention: Towards the underlying neural circuitry. In *Human Neurobiology, Springer-Verlag*, pages 219–227, 1985.
- [20] Laurent Itti, Christof Koch, and Ernst Niebur. A Model of Saliency-Based Visual Attention for Rapid Scene Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20:1254–1259, November 1998.
- [21] Maik Bollmann, Rainer Hoischen, and Bärbel Mertsching. Integration of static and dynamic scene features guiding visual attention. In *Mustererkennung 1997, 19th DAGM-Symposium*, pages 483–490, 1997.
- [22] Chia-Chiang Ho, Wen-Huang Cheng, Ting-Jian Pan, and Ja-Ling Wu. A User-Attention Based Focus Detection Framework and Its Applications. In *The 4th International Conference on Informations, Communications and Signal Processing and Fourth Pacific-Rim Conference on Multimedia*, volume 3, pages 1315–1319, 2003.

- [23] A. Ninassi, O. Le Meur, P. Le Callet, D. Barba, and A. Tirel. Task Impact on the Visual Attention in Subjective Image Quality Assessment. In *The 14th European Signal Processing Conference*, 2006.
- [24] Ming-Hsuan Yang, David J. Kriegman, and Narendra Ahuja. Detecting Faces in Images: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1:34–58, January 2002.
- [25] Jae-Beom Lee and Alexandros Eleftheriadis. Spatio-Temporal Model-Assisted Very Low-Bit-Rate Coding With Compatibility. *IEEE Transactions on Circuits and Systems for Video Technology*, 15:1517–1531, December 2005.
- [26] David Brown, Ian Craw, and Julian Lewthwaite. A SOM Based Approach to Skin Detection with Application in Real Time Systems. In *The British Machine Vision Conference*, 2001.
- [27] Vladimir Vezhnevets, Vassili Sazonov, and Alla Andreeva. A Survey on Pixel-Based Skin Color Detection Techniques. In *Graphicon-2003, Moscow, Russia*, pages 85–92, 2003.
- [28] Jörgen Ahlberg. A system for face localization and facial feature extraction. Technical report, Department of Electrical Engineering, Linköping University, Linöping, Sweden, 1999.
- [29] Y.H Chan and S.A.R Abu-Bakar. Face Detection System Based on Feature-Based Chrominance Colour Information. In *The International Conference on Computer Graphics, Imaging and Visualization*, pages 153–158, 2004.
- [30] J-C. Terrillon, M.N Shiraz, H. Fukamachi, and S. Akamatsu. Comparative performance of different skin chrominance models and chrominance spaces for the automatic detection of human faces in color images. In *The International Conference on Face and Gesture Recognition*, pages 54–61, 2000.
- [31] Li-Hong Zhao, Xiao-Lin Sun, Ji-Hong Liu, and Xin-He Xu. Face Detection based on Skin Color. In *The 3rd International Conference on Machine Learning and Cybernetics*, pages 3625–3628, 2004.
- [32] Yao-Xin Lv, Zhi-Qiang Liu, and Xiang-Hua Zhu. Real-Time Face Detection based on Skin-Color Model and Morphological filters. In *The 2nd International Conference on Machine Learning and Cybernetics*, pages 3203–3207, 2003.
- [33] Ming-Hsuan Yang and Narendra Ahuja. Gaussian mixture model for human skin color and its application in image and video databases. In *The SPIE:Conference on Storage and Retrieval for Image and Video Databases*, pages 458–466, 1999.

- [34] Qian Chen, Haiyuan Wu, and Masahiko Yachida. Face Detection by Fuzzy Pattern Matching. In *The International Conference on Computer Vision*, pages 591–595, 1995.
- [35] Michael J. Jones and James M. Rehg. Statistical Color Models with Application to Skin Detection. In *The International Conference on Computer Vision and Pattern Recognition*, pages 274–280, 1999.
- [36] Son Lam Phung, Douglas Chai, and Abdelassam Bouzerdoun. Adaptive Skin Segmentation in Color Images. In *The 3rd International Conference on Machine Learning and Cybernetics*, pages 3625–3628, 2004.
- [37] Son Lam Phung, Douglas Chai, and Abdelassam Bouzerdoun. Skin Segmentation Using Color Pixel Classification: Analysis and Comparision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1:148–154, Januari 2005.
- [38] Qiang Zhu, Kwang-Ting Cheng, Ching-Tung Wu, and Yi-Leh Wu. Adaptive Learning of an Accurate Skin-Color Model. In *The 6th IEEE International Conference on Automatic Face and Gesture Recognition*, 2004.
- [39] Rein-Lien Hsu, Mohamed Abdel-Mottaleb, and Anil K. Jain. Face Detection in Color Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 5:696–706, May 2002.
- [40] Javier Ruiz del Solar and Rodrigo Vershae. Skin Detection using Neighborhood Information. In *The 6th IEEE International Conference on Automatic Face and Gesture Recognition*, 2004.
- [41] Douglas Chai, King N. Ngan, and Abdesselam Bouzerdoun. Foreground/Background Bit Allocation for Region-of-Interest Coding. In *2000 IEEE International Conference on Image Processing*, volume 2, pages 923–926, 2000.
- [42] Scott Daly, Kristine Matthews, and Jordi Ribas-Corbera. Face-Based Visually-Optimized Image Sequence Coding. In *1998 IEEE International Conference on Image Processing*, volume 3, pages 443–447, 1998.
- [43] Somnath Sengupta, Shiv K. Gupta, and John M. Hannah. Percetually Motivated Bit-Allocation for H.264 Encoded Video Sequences. In *2003 IEEE International Conference on Image Processing*, volume 3, pages 797–800, 2003.
- [44] X.K. Yang, W.S. Lin, Z.K. Lu, X. Lin, S. Rahrda, E.P. Ong, and S.S Yao. Local Visual Perceptual Clues and its use in Videophone Rate Control. In *The 2004 International Symposium on Circuits and Systems*, volume 3, pages 805–808, 2004.

- [45] Trio Adiono, Tsuyoshi Isshiki, Kazuhito Ito, and Tomohiko Ohtsuka. Face Focus Coding under H.263+ Video Coding Standard. In *IEEE Asia-Pacific Conference on Circuits and Systems*, volume 3, pages 461–464, 2000.
- [46] Haohong Wang and Khaled El-Maleh. Joint Adaptive Background Skipping and Weighted Bit Allocation for Wireless Video Telephony. In *2005 IEEE International Conference on Wireless Networks, Communications and Mobile Computing*, pages 1243–1248, 2005.
- [47] Jeong-Woo Lee, Anthony Vetro, Yao Wang, and Yo-Sung Ho. Bit Allocation for MPEG-4 Video Coding With Spatio-Temporal Tradeoffs. *IEEE Transactions on Circuits and Systems for Video Technology*, 13:488–502, June 2003.
- [48] Jacob Augustine, Shrivarama K. Rao, Norman P. Jouppi, and Subu Iyer. Region of Interest Editing of MPEG-2 Video Streams in the Compressed Domain. In *2004 IEEE International Conference on Multimedia and Expo*, pages 559–562, 2004.
- [49] Haohong Wang, Yi Liang, and Khaled El-Maleh. Real-Time Region-of-Interest Video Coding using Content-Adaptive Background Skipping with Dynamic Bit Reallocation. In *2006 IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 2, pages 45–48, 2006.
- [50] Miska M. Hannuksela, Ye-Kui Wang, and Moncef Gabbouj. Sub-Picture: ROI coding and unequal error protection. In *2002 IEEE International Conference on Image Processing*, volume 3, pages 537–540, 2002.
- [51] Sanghoon Lee, Chris Podilchuk, Vidhya Krishnan, and Alan C. Bovik. Foveation-Based Error Resilience and Unequal Error Protection over Mobile Networks. *VLSI Signal Processing*, 34:149–166, May-June 2003.
- [52] Miska M. Hannuksela, Ye-Kui Wang, and Moncef Gabbouj. Isolated Regions in Video Coding. *IEEE Transactions on Multimedia*, 6:259–267, April 2004.
- [53] Chia-Wen Lin, Yung-Chang Chen, and Ming-Ting Sun. Dynamic Region of Interest Transcoding for Multipoint Video Conferencing. *IEEE Transactions on Circuits and Systems for Video Technology*, 13:982–992, October 2003.
- [54] Safak Dogan, Abdul H. Sadka, and Ahmet M. Kondo. Fast Region of Interest Selection in the Transform Domain for Video Transcoding. In *The 6th International Workshop on Image Analysis for Multimedia Interactive Services*, 2005.
- [55] ANSI T1.80103-1996. Digital transport of one-way video signals parameters for objective performance assessment. American National Standards Institute, 1996.

- [56] Zhou Wang, Hamid R. Sheikh, and Alan C. Bovik. Chapter 41 in the handbook of video databases: Design and application. B. Furth and O. Marqure, CRC Press, pp 1041-1078, 2003.
- [57] Sanghoon Lee, Marios S. Pattichis, and Alan C. Bovik. Foveated video quality assessment. *IEEE Transactions on Multimedia*, 4:129–132, March 2002.
- [58] Christian J. van den Branden Lambrecht, Oliver Versheure, Jerome Urbain, and Florent Tassin. Perceptual quality measure using a spatio-temporal model of the human visual system. In *SPIE*, volume 2668, pages 450–461, 1996.
- [59] Zhou Wang, Alan C. Bovik, and Ligang Lu. Why is image quality assessment so difficult. In *2002 IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 4, pages 3313–3317, 2002.
- [60] ITU-R Rec. BT.500-11. Methodology for the subjective assessment of the quality of television pictures. International Telecommunication Union, 2002.
- [61] MPEG-2 Codec. ffmpeg. <http://www.erightssoft.com/SUPER.html>, Retrived: 2006-06-10.
- [62] H.264/AVC Codec. Jm 10.1. <http://iphome.hhi.de/suehring/tml>, Retrieved: 2005-10-22.

Appendix A

Parametric skin detection model

In [32] Lv et al have suggested a simple elliptic two-dimensional model, which is based on the more complex model suggested by Hsu et al in [39]. In [39] it was shown that skin-color forms an elliptic cluster in the CbCr chrominance plane assuming that the YCbCr color space (See section 2.1.2) is used. Therefore, Lv et al in [32] apply the parameterized ellipse

$$E_p(m, n) = \frac{(C_r^{(f, (m, n))} - ecx)^2}{a^2} + \frac{(C_b^{(f, (m, n))} - ecy)^2}{b^2}$$

to indicate if pixel (m, n) in frame f belongs to a skin region in the CbCr plane, where $C_r^{(f, (m, n))}$ and $C_b^{(f, (m, n))}$ are the chrominance components of the pixel value $I^{(f, (m, n))}$. The constants are defined in [32] as $ecx = 1.60$, $ecy = 2.41$, $a = 1.60$ and $b = 14.03$.

Appendix B

$PSNR_{Border,Avg}$ for SP filtering

In the figures B.1 and B.2 the $PSNR_{Border,Avg}$ is plotted for various σ_1 of the SP filter, number of filtered and video sequences coded using H.264.

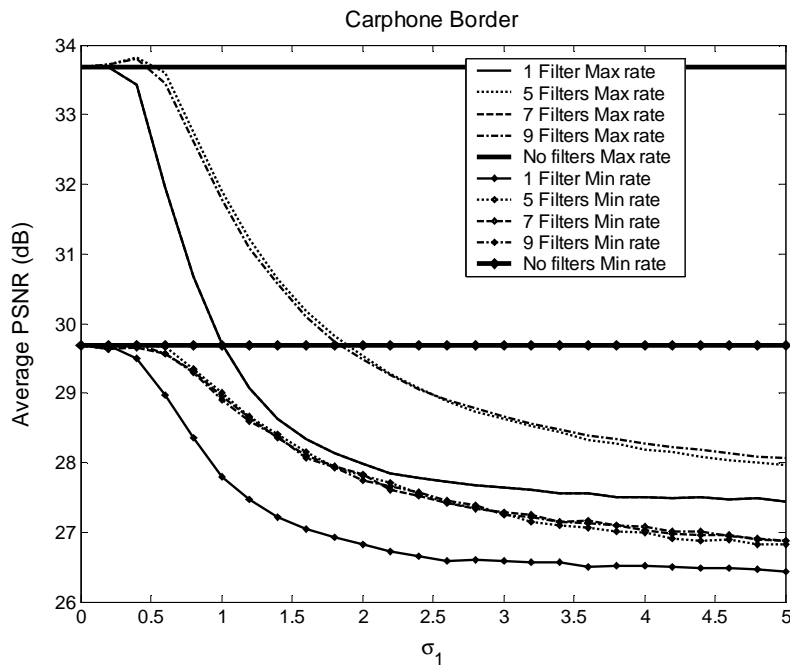


Figure B.1: The $PSNR_{Border,Avg}$ for different values of σ_1 and different numbers of gaussian filters are presented for the carphone sequence.

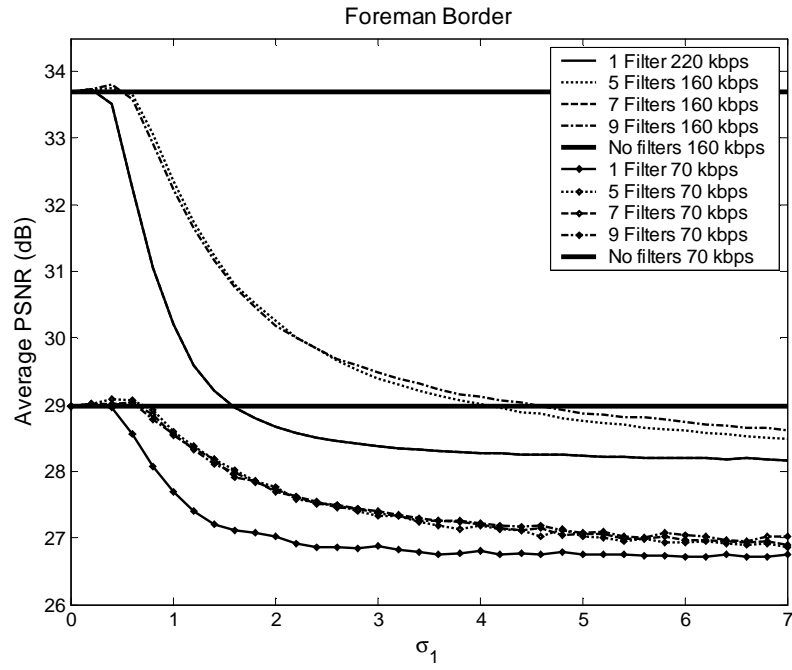


Figure B.2: The $PSNR_{Border,Avg}$ for different values of σ_1 and different numbers of gaussian filters are presented for the foreman sequence.

Appendix C

The α parameter

A short test was performed to verify the assumption that $\alpha = Q^{(m,n)}/A$ is a good choice for the α parameter in the bilinear filtering in section 4.1. The three different versions of α presented in figure C.1 are applied in the tests, where $\alpha = (Q^{(m,n)}/A)^2$ gives a higher impact of the pixel (m, n) in the current frame in most of the transition region. On the other hand, $\alpha = \sqrt{Q^{(m,n)}/A}$ gives a higher impact of the previous frame within the transition region and $\alpha = Q^{(m,n)}/A$ is a compromise of the previous two versions. The tests show (See figure C.2) that $\alpha = Q^{(m,n)}/A$ in average gives the best result of the three considering both the $PSNR_{ROI, Avg}$ and the $PSNR_{Border, Avg}$. The two other alternatives favor either $PSNR_{ROI, Avg}$ or $PSNR_{Border, Avg}$ and thus $\alpha = Q^{(m,n)}/A$ is applied in the tests.

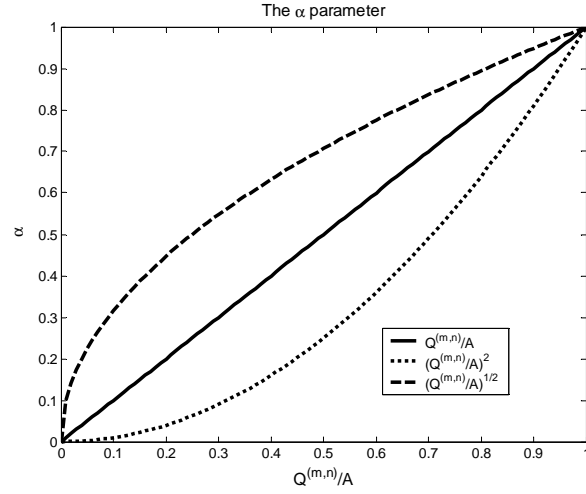


Figure C.1: The three different versions of the α parameter applied in the tests.

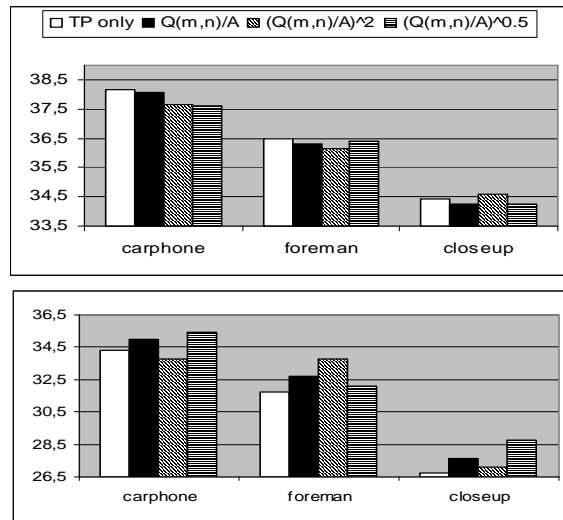


Figure C.2: Tests on the alpha parameter for three different sequences encoded with H.264 at 64 kbps gives $PSNR_{ROI,Avg}$ (top) and $PSNR_{Border,Avg}$ (bottom).

Biography

Linda S. Karlsson was born on the 4th of April 1976 in Katrineholm, Sweden. She received the Master of Science in Media technology and and engineering from Linköping University, Sweden in September 2002. During the studies before the university she had the opportunity to enroll as a student at Robert Johnson High School in Gainesville, Georgia in the USA in 1993–1994. She has also attended a year of exchange studies at the Rheinisch-Westfälische Technische Hochschule in Aachen, Germany in 2000–2001.

Karlsson is currently a PhD Student at the department of Information Technology and Media at Mid Sweden University. Her main research interest are in the field of video analysis and video source coding. Currently she is working on standard and codec independent region-of-interest video coding.