Video Decoder Concealment

Guillermo Arroyo Gomez
Abstract

Video Decoder Concealment

Guillermo Arroyo Gomez

H.264 is a new video coding standard developed jointly by ITU – T and the Moving Picture Expert Group (MPEG). It outperforms MPEG-2 by requiring around 50% of the bit rate for a similar perceptual quality. It allows low bitrate networks to supply video at a quality previously not possible.

Much of the previous work on concealment focuses on losses on the block level and uses the surrounding area information which is highly correlated. This thesis on the other hand analyzes and proposes ways to conceal errors in areas affecting much larger areas of a frame and with little information available from neighboring macroblocks.

The methods proposed in this thesis include using weighted interpolation to create the visual effect of smoothness for the user when doing spatial concealment. High frequencies are also removed from concealed areas after inter concealment using a spatial filter. A background detector is proposed to reduce background blurriness and a method is proposed to dynamically adjust the range of pixel values after the concealment is done. This thesis work introduces the use of a scene change detector when doing interframe concealment to avoid mixing two different scenes.

The results show that the perceived video quality can be significantly improved partly by removing highly noticeable artifacts and partly by giving a smooth image.
Acknowledgments

I would like to thank to all the people at Ericsson Multimedia Reasearch in the Visual Technology group for all the support I got from them during the thesis. Special thanks to my supervisor at Ericsson, Clinton Priddle for all the guidance, feedback, support and time for discussing ideas. I would like to thank Jonatan Samuelsson for his valuable help troubleshooting bugs in the code. Thanks to my reviewer Mikael Sternad at Uppsala University for his helpful ideas and corrections for this report.

I would also like to thank my parents and my friends for all the support I got from them while doing my masters degree.
Glossary

Blu-Ray – High Density disc use to store digital information including high-definition video.
CABAC - Context-adaptive binary arithmetic coding
CABLR - Content Adaptive Block Loss Recovery
CAVLC - Content Adaptive Variable Length Coding
CIF - Common Intermediate Format
CRT - Cathode Ray Tube.
DCT – Discrete Cosine Transform
DMVE - Decoder Motion-Vector Estimation Algorithm
DVD – Digital Video Disc
FEC – Forward Error Correction
FMO – Flexible Macroblock Order
H.264 – Video compression standard also know as MPEG-4 Part 10, or MPEG-4 AVC.
HD DVD - High Definition DVD
HDTV – High Definition Television
IPTV – Internet Protocol Television
JVT – Joint Venture Team
NAL – Network Abstraction Layer
MB – Macro Block
MSE - Mean Square Error
MPEG – Motion Picture Expert Group
MV – Motion Vector
NTSC - National Television System Committee
NAL – Network Abstraction Layer
RGB – Red Green Blue (Color Space)
SAD - Sum of Absolute Differences
SDTV - Standard Definition Television
PAL - Phase Alternating Line
PSNR - Peak Signal-to-Noise Ratio
VCEG - Video Coding Experts Group
VLC – Variable Length Coding
YUV – Color Space that stores the Luminace and Chroma information of a pixel.
1 Introduction

1.1 Motivation

As equipment for the consumer has become more sophisticated, a wider variety of multimedia applications are now being supported. The trend of continuously falling prices have helped consumers to start experiencing digital video in many ways such as DVD, HD DVD, Blu-Ray, SDTV, HDTV, Mobile TV and IPTV.

Some of these applications have large storage or broadband capabilities to carry video in high quality mode. They may have error resilience features such as being able to request the affected area from the source.

Other applications have far simpler error resilient features, such as Forward Error Correction (FEC). These tools work up to certain error rate. Above that rate they do not work.

Figure 1.1 illustrates this effect.

Figure 1.1 Left: Original image with no errors. Right: Shows the same image with packet loss. These gaps are clearly noticeable for the viewer. In order to reduce the impact on the visual quality perception, concealment in the affected regions needs to be done.

Due to the nature of video compression, packet losses do not only affect an area in the spatial dimension but the error propagates in future frames that use it as reference as well. The problem is harder in low bit networks, since one packet loss represent a bigger missing area.

The aim of this thesis is to study and optimize video decoding in lossy networks by concealing errors as much as possible. It will focus on packet losses in low bitrate streams which usually affect a significant area of the image.
2 Background

A video clip can be thought as a sequence of images being replaced at a certain rate of time giving the viewer the illusion of movement within the picture. Each individual image is known as ‘frame’. The frequency at which the frames are displayed is called the ‘frame rate’ and the common measurement unit is ‘frames per second’ (fps). In frames, the smallest element is the pixel. Frame resolution refers to the number of columns and rows of pixels that compose the picture (e.g., 640×480 or 1280×720).

Uncompressed video demands huge processing and storage capabilities. DVDs have 720x480 pixels at 24fps. Uncompressed RGB video is 3 bytes per pixel. So a 90 minute DVD movie in uncompressed RGB would come to 134GB.

Instead of using raw video, some compression techniques are used to deal with the storage problem. Natural video sequences usually contain areas that are highly correlated both in consecutive frames and spatially (Figure 1.2). Compression techniques utilize this redundancy. Most compression is lossy, i.e. it discards information while still keeping the result close to the original.

*Figure 1.2 Spatial and temporal correlation in a video sequence*
2.1 Color spaces

2.1.1 RGB
The RGB color space represents the color information of a pixel in three colors Red, Green and Blue. Once they are added together they can represent a wide range of colors. They are example of what is called Additive Primaries which create the sensation of a range of colors when they are combined together.

2.1.2 YUV
The YUV color space takes advantage that the Human Visual System is more sensitive to the position of brightness (luminance) than color (chrominance). It is represented with a luminance component (Luma) and two color difference components (Chromas). By giving more detail to luminance than color the bandwidth can be optimized. This can be done by sampling color at lower rates and no perceptible loss is incurred at normal viewing distances.

The luminance can be calculated by a weighted average of the R, G and B values in the RGB color space

\[ Y = K_r R + K_g G + K_b B \]

where Kr, Kg and Kb are the weighting factors.

The weighting factors Kr = 0.299, Kg = 0.587 and Kb=0.114 are used for standard television, which give us the following equations:

\[ Y = 0.299R + 0.587G + 0.114B \]
\[ C_r = 0.713(R - Y) \]
\[ C_b = 0.564(B - Y) \]

2.2 Interlaced video
In interlaced video the lines that compose a frame are scanned alternately. The set of lines scanned at any given time are called fields. Each field in a video sequence is sampled at a different time, the video signal’s field rate. In a Cathode Ray Tube (CRT), every field contains every second row or line of the image to be display, this is called interlacing. The next pass is performed on the gaps that were left behind in the previous one, this process is continuously repeated. This is carried from the top left corner to the bottom right corner of a CRT display. The afterglow of the phosphor in the CRT, in combination with persistence of vision is what makes the two fields being perceived as a continuous image.
2.3 Progressive video

Progressive video frames are transmitted, stored and displayed sequentially. The advantage of progressive video over interlaced video is that the problem of dealing with the temporal difference of all fields is eliminated. The disadvantage of progressive video is that it requires higher bandwidth than interlaced video of the same display resolution.

2.4 4:4:4, 4:2:2 and 4:2:0 YUV sampling

Figure 2.1 shows three different sampling formats.

![Figure 2.1. Different YUV sampling formats](image)

In format 4:4:4, all components have the same resolution. Since the human visual system is more sensitive to luminance than chrominance formats such as 4:2:2 and 4:2:0 are widely used. The chroma components in YUV 4:2:0 have quarter the resolution of the luma component. This format is commonly used in DVD and digital television.

2.5 Video frame formats

In television and other systems dealing with video, different frame formats are used. PAL is the standard used in Europe and NTSC in the USA. In order to make the transition among these formats the Common Intermediate Format (CIF) was invented. It has some variations, for example a quarter of the CIF resolution QCIF can be used which is only half of the height and width.

<table>
<thead>
<tr>
<th>Format</th>
<th>Video Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>SQCIF</td>
<td>128x96</td>
</tr>
<tr>
<td>QCIF</td>
<td>176x144</td>
</tr>
<tr>
<td>CIF</td>
<td>352x288</td>
</tr>
<tr>
<td>4CIF</td>
<td>704x576</td>
</tr>
<tr>
<td>16CIF</td>
<td>1408x1152</td>
</tr>
</tbody>
</table>

*Table 2.1 Video frame formats*
2.6 Human Visual Perception

Measuring the quality of a video sequence has proven to be a subjective task rather than an objective one. Visual quality from the viewer’s point of view “can depend very much on the task at hand, such as passively watching a DVD movie, actively participating in a videoconference, communicating using sign language or trying to identify a person in a surveillance video scene”[1].

Video engineers often use what is called Peak Signal-to-Noise Ratio (PSNR) to measure the quality of a video sequence. It can be calculated with the following formula:

\[ \text{PSNR}_{\text{dB}} = 20 \log \frac{255}{\text{MSE}} \]

where MSE is the Mean Square Error which is calculated by averaging the sum of squared differences between the current frame and the reconstructed frame. When MSE equals zero the PSNR is infinite.

It has its limitation since it doesn’t correlate with what humans perceive as quality. For example take a look at the next images:
All images besides the original have similar PSNR. Perceptually, the image with reduced color levels has the best quality but according to its PSNR it has the worst quality.

Another example is when packet losses occur. The viewer doesn’t know what visual information was supposed to be in the area affected by the error. In this case, it is more important that the affected area is concealed with something that is not noticeable for the viewer than that the values are similar to the ones of the original sequence.

### 2.7 Video Codec

A video codec (Figure 2.3) encodes a video sequence into a compressed stream and decodes back the compressed stream into a copy or an approximation of the original
sequence. If the image is identical to the original the process is lossless; if it differs from the original, the process is lossy.

![Diagram of Encoder/Decoder](image)

**Figure 2.3 Encoder/Decoder**

### 2.7.1 Encoder

A prediction is formed for each macroblock based on a reconstructed frame and the difference between the prediction and the reconstructed frame is encoded. The macroblock is encoded in intra or inter mode. In intra mode, the prediction is formed from parts from the same frame that already have been reconstructed. In inter mode, the prediction is created from previous encoded frames by shifting samples in the picture used for prediction. The picture used for prediction is called reference frame. The prediction is subtracted from the macroblock giving back a residual (Figure 2.4). A discrete cosine transform (DCT) is applied to the residual and then quantized. They are re-ordered, entropy coded and encapsulated along with other necessary information to decode the macroblock.

![Residual from ”Foreman” sequence](image)

**Figure 2.4 Residual from the ”Foreman” sequence**
2.7.2 Decoder
The decoder receives compressed video data contained in a Network Abstraction Layer (NAL). It is entropy decoded and reordered to give a set of quantized coefficients which then are rescaled and inverse transformed to get the residual. The prediction is added to the residual to create a decoded macroblock.

2.7.3 Motion Estimation and Compensation
Instead of encoding a signal from scratch the encoder can use a previously encoded signal as a prediction. If the prediction is good then the residual between the prediction and the reconstructed frame should be small. The process of finding how pixels values have moved from previous frames is called motion estimation. It is usually performed on a block by block basis, i.e. each frame is divided into blocks of pixels and each block uses pixels in a reference frame for prediction. The shift in pixels with the best match is called motion vector (MV). The amount of prediction error can be measured using the mean squared error (MSE) or sum of absolute differences (SAD) between the actual pixel and the predicted pixel values for the motion compensated region.

Block based motion compensation is done when decoding or reconstructing a frame by applying the displacement described in motion vectors to the current macroblocks.

2.7.4 Transform and Quantization
A transformation is applied to the residual in order to represent the data in a more efficient way. It doesn’t compress any data but it will help to remove spatial correlation and concentrate energy in only a few coefficients. The transform is reversible so the inverse transform converts the coefficients back to the spatial domain.

The most common used transformation in video coding is the two dimensional discrete cosine transform (2D-DCT). It is applied on blocks of pixels instead of the entire image. 2D-DCT is implemented by using matrix multiplication.

Coefficients obtained by the transform can have very different values making entropy coding difficult. By rounding the value of coefficients to certain levels, i.e. quantization of transform coefficients, less significant coefficients can be discarded by making them zero and they are no longer necessary to transmit or store.

2.7.5 Entropy Coding
When transmitting quantized coefficients, more compression can be done by removing statistical redundancy. Entropy coding is a lossless process, i.e. no information is lost.

2.7.5.1 Huffman coding
Huffman coding uses a variable length coding table (VLC) for encoding a symbol. The VLC table is derived based on the probability of occurrence of the source symbol. Shorter codes are used for the most common occurring symbols.
2.7.5.2 Arithmetic coding

In arithmetic coding a sequence of symbols is represented as an interval. The probability of occurrences for each symbol must be known to create this interval. Arithmetic coding compresses data more efficiently than Huffman coding but requires more processing power to decode.

2.8 H.264 overview

H.264, also known as MPEG 4 part 10 or MPEG 4 AVC, is a standard for video compression [11]. It was written as a collaborative effort between the Video Coding Expert Group (VCEG) and the Moving Picture Experts Group (MPEG) as a product of a partnership known as the Joint Video Team (JVT). The H.264 standard documents two things - a syntax that describes visual data in a compressed form and a way of decoding the syntax to reconstruct visual information [1].

In H.264, a frame is divided into blocks of 16x16 pixels called macroblocks (MBs). A set of macroblocks is grouped into arbitrary shaped slices (Figure 2.5) and then encapsulated into a NAL.

Figure 2.5 Two slices in a QCIF frame

There are five types of slices (Table 2.2 [1]).
<table>
<thead>
<tr>
<th>Slice type</th>
<th>Description</th>
<th>Profile(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>I (Intra)</td>
<td>The slice contains only Intra macroblocks.</td>
<td>All</td>
</tr>
<tr>
<td>P (Predictive)</td>
<td>The slice contains Inter (Predictive from previous frames) and/or Intra macroblocks types.</td>
<td>All</td>
</tr>
<tr>
<td>B (Bi-predictive)</td>
<td>The slice contains Bi-predictive (Predictive from previous and future frames) and/or Intra macroblocks types.</td>
<td>Extended and Main</td>
</tr>
<tr>
<td>SP (Switching P)</td>
<td>Facilitates switching between different precoded pictures.</td>
<td>Extended</td>
</tr>
<tr>
<td>SI (Switching I)</td>
<td>Facilitates switching between precoded bitstreams.</td>
<td>Extended</td>
</tr>
</tbody>
</table>

*Table 2.2 H.264 slices modes*

For a more detailed discussion about B, SI and SP slices types refer to [1].

The number of macroblocks per slice varies from one macroblock to the total number of macroblocks in a picture. No slice is shared between two frames and each slice has minimal interdependency between coded slices helping to reduce the propagation of errors.

Macroblocks can be subdivided further into small blocks called partitions. For intra frames two different sizes can be use 16x16 and 4x4. Inter frame macroblocks can be divided into partitions: 16x8, 8x16, 8x8. Every 8x8 can be further divided into subpartitions: 8x4, 4x8, 4x4. These are illustrated in the following figure.

*Figure 2.6 Macroblock and subblock partitions. 8x8 MBs can be divided into subblock partitions.*
2.8.1 Quarter-pixel motion estimation and motion compensation

Quarter-pixel is used in H.264 to get a higher precision by allowing the motion vector to be non-integer number and allow the shift to be down to a quarter of a pixel (Fig. 2.7).

H.264 uses a 6-tap FIR filter for half-pixel interpolation and then a simple bilinear filter to achieve quarter-pixel precision from the half-pixel data. The encoder is able to calculate the halfpixel-interpolated frame before the encoding process, while the quarter-pixel data is calculated only when needed.

2.8.2 Flexible Macroblock Ordering (FMO)

H.264 allows the division of an image into regions called slices groups. FMO is a way of grouping macroblocks into slices. FMO consist of 7 different types (Figure 2.8). They are labeled from 0 to 6 with type 6 being the most random and allowing full flexibility.

Type 0: Uses a fixed run length for each slice. They are repeated until it fills the frame.
Type 1: Uses a mathematical function to scatter the slices.
Type 2: Marks rectangular areas called regions of interest.
Type 3-5: Lets the slice groups grow and shrink over the different pictures in a cyclic way.

Figure 2.7. Quarter pixel values.
FMO is considered an error resilience feature. If a slice is lost then the remaining macroblocks of error free slices can help to conceal the missing slice. For example in type 1, if a slice is lost spatial interpolation can be done to conceal the missing macroblocks.
2.8.3 Profile and levels
Profiles specify the syntax (i.e. algorithms) and levels specify various parameters (resolution, frame rate, bit-rate, etc.). All profiles support I and P slices types, ¼-pixel motion compensation and Content Adaptive Variable Length Coding (CAVLC).

2.8.3.1 Baseline Profile (BP)
Baseline profile is designated for low cost applications with limited computing resources such as video conferencing, video-over-IP, and mobile applications. Tools used by Baseline profile include [2]:
- Arbitrary slice ordering (ASO)
- Flexible macroblock ordering (FMO)
- Redundant slices (RS)
- 4:2:0 YUV Format

2.8.3.2 Extended Profile (XP)
Extended profile is intended as the streaming video profile. It supports:
- B, SI and SP slices.
- Slice data partitioning
- Weighted prediction
- Arbitrary slice ordering (ASO)
- Flexible macroblock ordering (FMO)
- Redundant slices (RS)

2.8.3.3 Main Profile (MP)
Main profile is intended for a wide range of broadcast and storage application. Tools supported:
- Interlaced coding
- B slice type
- Context-adaptive binary arithmetic coding (CABAC) entropy coding
- Weighted prediction
- 4:2:2 and 4:4:4 YUV, 10- and 12-bit formats

2.8.3.4 High Profile (HP)
Intended for broadcast and disc storage applications, particularly for high-definition television. It adds support for adaptive selection between 4x4 and 8x8 block sizes for the luma spatial transform and encoder-specified frequency dependent scaling matrices for transform coefficients.
3 Previous work

Most of the previous work done in this field assumes that only one macroblock or row of macroblock are lost so they have immediate surrounding spatial information available for the concealment.

The Boundary Matching Algorithm (BMA [3]) exploits the fact that adjacent pixels in a video frame have high spatial correlation. It takes the lines of pixels above, below, and to the left of the lost macroblock in the current picture and uses them to surround each candidate from the previous picture. BMA then calculates the total square difference between these three lines with the edges of each candidate macroblock in the previous decoded picture. The motion vector is calculated based on which the squared difference between the surrounding lines and the block from the previous pictures is a minimum.

The Decoder Motion-Vector Estimation Algorithm (DMVE [4]) like BMA also exploits temporal information around the lost macroblock. Instead of just the lines of pixels above, below, and to the left of the lost macroblock it includes the above-left and bottom-left (if received correctly). If none of these macroblocks were received correctly then it uses them after they are concealed. It performs a full search within the previous picture for the best match of lines surrounding the missing macroblock. DMVE can consider up to 16 lines encircling the macroblock lost.

The Content-Based Adaptive Block Loss Recovery (CABLR [9]) uses temporal image information for macroblock loss recovery, if temporal information fits well. Otherwise correctly received or concealed spatial neighboring macroblocks are used to recover a lost macroblock. Finally a range constraint is applied on the spatially recovered macroblock.

H.26L error concealment in [7] proposes to use two different algorithms for intra frame concealment and inter frame concealment. Lost areas in Intra frame can be concealed spatially by doing a weighted pixel averaging. The weights used are the inverse distance between the source and destination pixels. Only correct neighboring macroblocks are considered if at least two are present, otherwise concealed macroblock are used. In Inter frame concealment, the motion vector of the lost macroblock is predicted from a neighboring macroblock relying on the fact that motion of neighboring areas is often highly correlated. The motion vector that results in the smallest luminance change across boundaries when the macroblock is copied in the frame is selected.

Spatio-Temporal Fading Scheme for Error Concealment in Block-Based Video Decoding System is based on a boundary error criterion obtained from temporal error concealment, either spatial, temporal, fading or a combination of these. The weights for fading are interpolated from the boundary error. The boundary error is represented as a weighted absolute difference between well received macroblock boundary samples from the current frame and motion compensated macroblock boundary sample from the previous frame [5].
Other concealment methods, such as [6][8], use simple spatial interpolation assuming that neighboring macroblocks are available.

Methods from previous work are more suitable for high bitrate applications such as HDTV where the loss of a packet usually represents one macroblock or non-continuous macroblock lines of the frame. They rely on using the pixels surrounding the missing area to try to match that boundary with previous frames and use that information to reconstruct the missing macroblock. Therefore, methods from previous work are not suitable for applications where little information is available.

A direct comparison between the previous work methods and the ones proposed in this thesis is difficult to make. This work is focused on low bitrate networks and assumes that little information is available, part of the guessed motion vectors in the continuous frame area don’t have correlation with the real ones and the information from the macroblock residual is lost. Thus, emphasis is put on reducing the visual impact of the artifacts that might arise in the frames following the concealed frame.
4 Proposed Methods

This section describes the approach used for the proposed methods, how to induce errors by dropping packet of the bitstream, a small overview of the test sequences used and methods used for intra concealment and inter concealment.

4.1 Proposed method approach

The approach of the proposed methods in this thesis work uses a low bitrate model such as mobile TV. The thesis focuses entirely on packet losses. If a packet arrives with bit errors then it cannot be trusted and it is completely discarded. Such packet loss usually results in losing an entire frame or a large part of a frame. This work will be based on the base line profile and FMO will not be considered.

4.2 Dropping Packets

Errors were induced by droppingNALs from the bitstream. Thus, the decoder will detect the absence of that NAL and call the corresponding function to perform some sort of concealment. At first this was done on selected areas of specific frames where there was suspicion that strange behavior will be observed. Once the concealment tools were tested enough on those sequences then the induction of errors was random.

4.3 Test Sequences

Several video sequences were used to test the proposed concealment methods. Most of the following sequences are freely available on the web [11] for research purposes. A brief description about the motion and static areas is given below.

Carphone – This sequence shows a person inside a car. Motion in the scene is shown at the window of the car, facial gestures and hands movement of the person talking.

Coastguard – The upper half of the frame doesn’t show up significant motion. In the bottom half two boats move in opposite directions. The flow of water also represents significant motion.

Bus – A TV station logo is in a fixed position at the bottom right part of the frame. Panning and zooming is used to keep track of a bus through a street moving in a horizontal direction.

Container – A cargo container and a small boat represent most of the motion in this scene by moving slowly from left to right at the upper part of the frame. Water ripples give no significant motion to the lower part of the frame. Some birds pass across the screen at the end of the sequence.
Flower – A windmill and some people walking at the center of the initial frames give some motion to the scene. The camera is panning to the right hand side for the rest of the scene showing up a garden in the bottom part of the scene.

Foreman – A very popular sequence among video engineers. It shows a construction worker moving his head and making facial gestures to the camera. The second half of the sequence shows a construction site next to the construction worker. During the sequence the camera presents some tilting.

Hall Monitor – The camera is at a still position all the time recording a hall in an office. Some motion is presented in the middle region of the frames when two workers walk along the corridor in opposite directions. One of them drops a suitcase; the other one grabs a small TV set.

Mobile – A small plastic train moves at the bottom of the screen from right to left pushing a ball. One calendar on the background does up and down motion. At the end of the train passes behind a toy rotating in several different angles.

Salesman – A salesman is at his desktop producing some motion with his head, and hands which hold a rectangular object. The background remains with no motion and the bottom part of the frames shows the shadows produced by his movements.

Stefan – A lot of motion is presented in this scene by keeping track of the tennis player in action. The background is in constant motion all the time due to the panning of the camera.

Brit Awards – 3 scene changes detected in this clip. The first scene and fourth scene have significant motion because the person in focus is moving and some people are moving around.

Shine – A person singing inside and outside a subway station. Lots of motion created by the people passing by. Several scene changes and areas with different brightness intensities. A complex scene to encode/decode.

### 4.4 Intra concealment

Concealment in Intra Frames relies only on spatial information from the current frame. Intra Frames are usually inserted when there’s a significant amount of energy needed to encode the current frame like in a scene change or when there are no reference frames available.

#### 4.4.1 Smooth macroblock mosaic

Smooth macroblock mosaic is a simple concealment method which gives a uniform color to the missing block by taking the mean color of the three macroblocks from above (Fig. 4.1b). The mean of the neighboring macroblocks from above (ABC) is calculated
individually by adding all the pixel values and dividing them by the total amount of pixels per macroblock.
The new color for the concealed macroblock is calculated by \((A + B + C)/3\)

When the missing macroblock is on the edge of the frame (Fig. 4.1a and Fig. 4.1c) then more weight is given to the macroblock that is directly above (or below in a special case) so that the color for the new concealed macroblock is \((2A + B)/3\) for fig. a.
A special case is made when no line from above is available then it uses the macroblocks from below or keeps scanning until it finds them.

Smooth macroblock mosaic is low complexity and gives better results than just displaying a uniform color. Figure 4.3 shows some results compared with macroblock concealed by green color.
4.4.2 Gaussian Weighted interpolation

Gaussian Weighted interpolation takes the idea from spatial filters that are applied usually to static images. The next figure shows a filter using the coefficients of a Gaussian distribution (mean=0, standard deviation=1).

![Figure 4.3. Smooth MB mosaic concealment compared with original frames and green MB concealed frames](image)

The pixel that is currently being evaluated is the one that corresponds to coefficient 41. All of these coefficients represent the weight to use when the calculation of the weighted average color value for that pixel is done.
This same technique can be used to interpolate and create a ‘smoothing’ effect even if the current pixel to evaluate is missing as long as some neighboring pixels are available.

Gaussian interpolation works at the pixel level by averaging the values of neighboring pixels and giving more weight to the pixels closer to the one that is the filter is currently being applied on.

The following filter assumes that you only have the upper pixel lines available.

<table>
<thead>
<tr>
<th>Normalization factor</th>
<th>1/83</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 4 7 4 1</td>
<td></td>
</tr>
<tr>
<td>4 16 26 16 4</td>
<td></td>
</tr>
<tr>
<td>x x x x x</td>
<td></td>
</tr>
<tr>
<td>x x x x x</td>
<td></td>
</tr>
<tr>
<td>x x x x x</td>
<td></td>
</tr>
</tbody>
</table>

*Table 4.2 Gaussian 2D distribution upper two lines*

The algorithm to use here should ignore pixels at positions where the coefficient is marked with an ‘x’.

The position of the current pixel to interpolate should be zero in this case.

An inverse filter can be used when the bottom pixel lines are available.

When there’s a gap of lines missing and information from the upper pixels and bottom pixels is available, two pass can be done with different filters to exploit the information available from both sides as illustrated in the next figure:
For each pass, the corresponding filter is used to calculate the value of the pixel at a given position. This value is then multiplied by a factor depending how much contribution it got from the side the filter started to get the information from.

### 4.4.3 Mean weighted interpolation

This method uses the same method as the Gaussian distribution, except that all the weights in the filter are set to 1.

The mean distribution eliminates frequencies which are not dominant in the area where it is applied to. Gaussian distribution on the other hand gives more weight to the current pixel which gives the effect of vertical lines since it only takes the pixel information from one end.
The size of kernels used also affect the resulting concealment. The bigger the size the more blurred the area becomes (Figure 4.5).

In the case of two way interpolation, this is very hard to see especially since both ends usually have different pixel values and the region in between is faded more evenly (Figure 4.6). After a couple of frames they were practically indistinguishable.

Figure 4.5. One way interpolation. Top left: Gauss Kernel=5x5. Top Right: Mean Kernel=5x5 Bottom Left: Gauss Kernel=7x7. Bottom Right: Mean Kernel=7x7
After testing done in several sequences the mean weighted interpolation with kernel size 7x7 was the one that selected for spatial weighted interpolation due to its evenly blurring in one way interpolation.

4.5 Inter concealment

Concealment done in Inter Frame relies on temporal and spatial information. The proposed method uses the motion vectors and reference picture information from macroblocks in the same frame. It might be natural to think that concealment in Inter Frames should always take information from the temporal dimension to do the concealment since the missing area might be similar to the one from last picture. However, if there’s a scene change or a huge amount of energy on the correctly received macroblocks then the temporal correlation is less so it’s preferable to use a concealment method that is based on spatial information only such as the ones used in the concealment done in Intra Frames.

When macroblocks are lost in a inter frame, what is lost are the motion vectors and the residual corresponding to that macroblock. A simple way to conceal those macroblocks is to copy the pixels from the previous frame at the same position. Unfortunately in most of
video sequences the lost motion vectors are not zero so they don’t refer to the same position in the previous frame. The next frame that uses the concealed frame as a reference frame assumes it is correct which might show significant artifacts.

Assuming that not much energy is lost in the residual, concealing only a macroblock in a frame would be relatively simple just by getting as much information from the available neighboring macroblocks as possible. The more correlation among the neighboring macroblocks, the better the concealment.

The problem increases when more than one row of macroblocks are lost since there might be less correlation the further away these macroblocks are located from the neighbors that arrived correctly. For example, consider a frame where the bottom half of the macroblocks are lost. In most of the observed video sequences, many of the motion vectors from the last row of macroblocks that successfully arrived have high correlation with the first row of missing macroblocks (Figure 4.8). In other words, they have similar motion vectors, but that's not usually the case with the last row of missing macroblocks.

---

**Figure 4.7** Macroblock missing in Frame 1. Pixels from previous frame copied.

---
Trying to predict the motion vector for each individual missing macroblock when there’s low correlation with the ones used for predicting might be counterproductive. By using different motion vectors values for each macroblock, the content might not form a consistent picture and artifacts in the border region of each concealed macroblock will be noticeable.

### 4.5.1 Proposed method

In order to keep a consistent image within the missing macroblock area, this method assigns a fixed motion vector and a fixed reference frame for all the missing macroblocks in a continuous area by analyzing the motion vectors of the row above the missing macroblocks (Figure 4.9). No residual is added so the referenced area is basically copied from the previous frame. If there aren’t any rows available above the missing macroblocks then it looks at the row below the missing macroblocks. If there aren’t any rows below the lost area then the motion vectors are set to zero and the reference frame is set to the previous frame.
The analysis done on the row of available macroblocks consists of comparing the motion vectors making sure they refer to the same reference frame and counting how many motion vectors correlate with the motion vector that’s being evaluated. Any motion vector within a specific range is classified as correlated. Values from 0.5 up to 3.0 were tested in 0.5 increments. A value of ±1.0 in the x and y axis gave the best results.

The image presented in the missing area will have consistency by presenting an uniform image copied from a previous frame and only the edges will present significant artifacts due to misalignment with the portion of the picture that was received correctly. Future frames that used the concealed frame as reference will contain artifacts depending on how good the predicted motion vectors were from the real ones and the residual information.

### 4.5.2 Scene change detector

There is a special case to consider when using concealment on inter-frames. The selected motion vector might be pointing to a reference picture before a scene change was done. In other words, the reference picture used to conceal the current frame might have content from a very different scene, therefore the resulting picture will show up two different scenes mixed together and artifacts in subsequent frames will be significant (Figure 4.10).

A proposed solution to deal with scene change is to use a scene change detection method. When a scene change is done, it is usually placed into an intraframe but it’s not always the case since that’s the decision of the encoder. If it’s placed at an interframe, it has many intra coded macroblocks and/or big residuals. In any case, the energy stored at the residual is considerably bigger than an inter frame belonging to the same scene. Even if
the scene doesn’t change completely as long as energy residuals change dramatically the scene it should be treated as scene change. So by looking at the length of the received slices and comparing them with the slices of the previous frame, it’s possible to detect a scene change. Another way to detect a scene change is by counting the amount of intra macroblocks in the neighboring slice.

Once a scene change is detected intraframe concealment can be use instead since the frame is changing dramatically in the temporal domain.
Figure 4.10. Top left: Reference frame. Top right: Original frame with no errors. Middle left: Interframe concealment. Middle right: Intraframe concealment. Bottom left: Interframe concealment after 23 frames. Bottom right: Interframe concealment after 23 frames.
High frequencies on the picture have big impact on the visual experience (Figure 4.11). When motion vector from future frames use a concealed picture as reference, high frequencies may induce artificial edges.

One way to lower the impact of high frequencies is to blur the concealed area. This is done by applying a spatial filter. The spatial filters tested were the Mean Filter and Gaussian Smoothing.

The mean filter replaces each pixel with the average value of its neighbors, including itself. This has the effect of eliminating pixel values that are unrepresentative of its surroundings.

\[
\begin{array}{cccccc}
1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1
\end{array}
\]

Gaussian smoothing works similar to the mean filter except that it uses a 2-D Gaussian distribution. It blurs the image as well but it gives a weighted average of each pixel’s neighborhood, giving more weight to the central pixels.

\[
\begin{array}{cccccc}
1 & 4 & 7 & 4 & 1 \\
4 & 16 & 26 & 16 & 4 \\
7 & 26 & 41 & 26 & 7 \\
4 & 16 & 26 & 16 & 4 \\
1 & 4 & 7 & 4 & 1
\end{array}
\]

The smoothing done by using Gaussian Smoothing is gentler and preserves edges better than a similar sized mean filter (Figure 4.12).
It will help future frames by keeping more information about the structure of objects and details so it removes some high frequencies while it doesn’t appear as a dramatic change for the viewer.

Figure 4.13 shows frames concealed using the proposed method for inter frames. The green color for spatial concealment was used as a reference to locate the affected area. A version with no filter and filter is shown at different frames.
Future work

The following ideas are conceptually discussed and but have not been not fully implemented. They require more time in order to be implemented than that available for this thesis work.

5.1 Dynamical range adjustment of pixel values

Different areas from future frames can be affected not only by a bad mismatch in the motion vectors used for the macroblocks in the missing area but also by the lack of information about the residual. One way to reduce this effect is to keep track of the macroblocks pointing or using information coming from a concealed area, for a certain interval of frames (Figure 5.1). The decoder should check the values for the pixels generated by these macroblocks, if they are out of the range (over 255 or below 0) or unrepresentative of the rest of the sequences then the values should be dynamically adjusted within the allowed range in order to give a better match.

![Figure 4.13. Concealment in inter frames.](image)
It can also be possible to do some statistical analysis on the macroblocks based on the range of values in several future frames affected by the same pixels in the missing area and try to deduce the range of original values for those pixels and then compensate the pixels affected by them.

### 5.2 Background detector

Blurring can be very helpful by removing high frequencies that might degrade the quality of future frames to decode but it also blurs areas that don’t have significant changes over time and can be better thought as background. Pixels in background areas tend to have a bigger life span than areas where motion is taking place thus avoiding unnecessary blurring will give a better visual experience to the viewer. A background detector (Figure 5.2) is proposed in order to deal with this situation.

The background detector can check the macroblocks of previous frames situated on the same edges of the current missing area. A good indicator to detect the background
area is to look in the motion vectors for 0 or SKIPPED values and see if the neighboring macroblocks have the same motion vector. Once that area is detected then the proper motion vector should be set and the smoothing filter can be skipped.
6 Conclusions

Concealment on relatively big frame areas is very complex. It is not only a problem of trying to predict motion vectors but also the loss of the residual information.

Several sequences were analyzed in order to get clues about the problems and possible solutions to implement. Sequences with not much motion tended to be easier to conceal since most of the motion vectors were correlated and little energy was lost in the residual. On the other hand, sequences with a lot of motion activity tended to be difficult to conceal initially but after several frames the residual of future frames tended to gradually correct the missing area. The sequences that proved harder to conceal were the ones where the missing area was part of the background and the assigned motion vectors to that area come from an active area. Since the area was static, it doesn’t converge to the correct values for the rest of the sequence.

Other methods from previous work described in chapter 3, concentrate their testing more at the macroblock level or one row of macroblocks trying to get as much information as possible from the surroundings since the correlation is high. The methods discussed in this thesis, focus on areas concerning two or more missing rows of macroblocks in order to model a packet loss in low bitrate networks. Emphasis was put on removing high frequencies that affect visual perception during the following frames due to the fact that only one motion vector was used for the whole missing area. A background detector was proposed to skip unnecessary blurring caused by the spatial filter that removes high frequencies. A dynamical range adjustment of pixel values to boost the convergence of wrong pixel values to the real ones after concealment is also proposed.

One major implemented feature, which methods from previous work in chapter 3 didn’t consider, was the use of a scene change detector that avoids the mixing of two different scenes when doing inter concealment.
7 References

1. Ian E.G. Richardson. H.264 and MPEG-4 Video compression. John Wiley & Sons Ltd. 2003