Weighted Combination of Sample Based and Block Based Intra Prediction in Video Coding

Signe Sidwall Thygesen

Master's thesis
2016:E2

LUND UNIVERSITY

Faculty of Engineering
Centre for Mathematical Sciences
Mathematics

CENTRUM SCIENTIARUM MATHEMATICARUM

LUNDS
UNIVERSITET

Lunds Tekniska Högskola

Signe Sidwall Thygesen

supervisors:
Per Hermansson, *Ericsson*
Kenneth Andersson, *Ericsson*
Karl Åström, *LTH*

Abstract

The latest standard within video compression, HEVC/H.265, was released during 2013 and provides a significant improvement from its predecessor AVC/H.264. However, with a constantly increasing demand for high definition video and streaming of large video files, there are still improvements that can be done. Difficult content in video sequences, for example smoke, leaves and water that moves irregularly, is being hard to predict and can be troublesome at the prediction stage in the video compression. In this thesis, carried out at *Ericsson* in Stockholm, the combination of sample based intra prediction (SBIP) and block based intra prediction (BBIP) is tested to see if it could improve the prediction of video sequences containing difficult content, here focusing on water. The combined methods are compared to HEVC intra prediction. All implementations have been done in Matlab. The results show that a combination reduces the Mean Squared Error (MSE) as well as could improve the Visual Information Fidelity (VIF) and the mean Structural Similarity (MSSIM). Moreover the visual quality was improved by more details and less blocking artefacts.

Contents

ABBREVIATIONS

IntraBC	Intra Block Copy
HEVC	High Efficiency Video Coding
PSNR	Peak Signal-to-Noise Ratio
MSE	Mean Squared Error
VIF	Visual Information Fidelity
SSIM	Structural Similarity
MSSIM	Mean Structural Similarity
HVS	Human Visual System
QP	Quantization Parameter
AVC	Advanced Video Coding
SBIP	sample based intra prediction
BBIP	block based intra prediction

Chapter 1

Introduction

The growing popularity of HD Video and an increasing amount of video services, such as video chatting and mobile video streaming, is creating a stronger need for better video compression [1]. Uncompressed video signals create huge amounts of data and at the same time video usage has become more and more available for everyday usage in video streaming and real-time video chatting. The hunger for greater access to video content is constantly increasing as well as the demand for higher quality video. This requires very good compression algorithms in order to transfer the data through the network, but still the video traffic is the biggest load on communication networks and data storage world-wide [2].

Especially some temporal dynamic structures, i.e. structures changing over time, could be hard for an encoder to predict which increases the data needed to compress that sequence. In this work the focus has been on water structures and how to improve the current prediction in the latest standard for such sequences.

This report is structured as as follows. First of all some related work will be discussed and the aim of the thesis. In chapter 2 then follows some relevant background information about video compression and more detailed about the intra prediction methods as well as video and image quality assessment. The proposed method is described in chapter 3 with and overview in 3.1. The results and a discussion about these is found in chapter 4 with following conclusions and future work in chapter 5.

1.1 Related work

This thesis is within the area of improving on intra prediction as well as connects to the area of texture synthesis within perceptual video coding. Here some previous related works are described in order to give an overview of the background.

1.1.1 Sample based intra prediction

The sample based intra prediction uses neighbouring pixels in a frame to predict a certain block (see section 2.2.1). The current standard within video coding, High Efficiency Video Coding (HEVC)/H.265, bases its intra prediction on a sample based intra prediction method. HEVC was officially released during 2013 and is a successor to the widely popular H.264 standard. It is expected that even better video codecs will be standardized in the future as the demand for on-line video continues to grow. The intra coding tend to transport most of the signal energy in the video stream, and therefore any improvements on the prediction and coding methods is important for the reduction of the bits needed when compressing a video sequence [3].

1.1.2 Block based intra prediction

Since the intra prediction is known for giving a poor result when predicting blocks with more complex structure and also that pixels that are far from the reference samples are usually badly predicted, methods of reusing information that is already encoded have been investigated. This has been proven efficient in several works, some of them listed here below.

Chrysafis et al proposed a method in 2002 where already encoded blocks within the same frame are copied and used as prediction for an unencoded block [4]. A search window is defined and the difference between the unencoded block and each possible block within the frame is computed. The block having the smallest difference is used to encode the block, together with the difference.

The method of reusing blocks is often called *Intra Block Copy* since it copies blocks within the same frame - the intra frame. Here the position from where the block is copied is saved as a block copy vector with the vertical and horizontal position. This is then coded and sent to the decoder so this knows from where it should copy.

For sending the choice of prediction, if it should be a block based and sample based intra prediction, Kadono et al proposed a method in 2004. This method is based on the previous standard H.264/AVC and is using a tree structure in the bitstream as mode information. The choice is based on Rate-Distortion Optimization. This method showed an improvement in video quality [5].

In a paper by Suzuki et al from 2006 the intra prediction is improved by using a so called *Template Matching* technique [6]. This method uses a template which consists of already encoded samples adjacent to the block when looking for the best matching block. The Template Matching is performed in both the encoder and decoder and thus allowing smaller blocksizes to be used without having to send a lot of extra information.

Ballé J. and Wien M. wrote in 2007 about an extension of the Intra prediction to exploit self-similar properties of the encoded texture [3]. The idea was similar to previous work, to imitate motion compensation and to match the current block with another within the frame and simply copy this, with the purpose to remove redundant data. They based it on the existing video compression standard H.264/AVC and could show that depending on the content of the video sequence, substantial gains in rate-distortion performance was achieved. Here they also referred to the Template Matching method.

The method of reusing blocks has also been proved efficient for screen content coding in 2013 and 2014 by Kwon D-K. and Budagavi M. [7], [8]. With screen content it means any sequences consisting of material produced by a computer screen, for example computer graphics. The screen content coding extension of HEVC is currently in draft stage and is expected to be finalized during 2016.

In a paper from Chen et al in 2015, the method is also applied to natural content video coding by using Template Matching [9].

1.1.3 Combining block based inter prediction with sample based intra prediction

Andersson K. proposed in 2006 a weighted combination of intra and inter prediction (CIIP) [10]. This is hence combining the spatial prediction from intra and the temporal prediction from inter in order to improve the prediction of the block. In the paper it is observed that this combination yields an improved performance to the previous standard H.265/AVC. The approach works best for difficult motion.

In a paper from Li J. et al a combination of intra and inter is further investigated by fine tuning the weighted coefficients [11]. To avoid additional signalling the correlation between the current block and the neighbouring blocks is analysed in order to optimize the weighting coefficients from the already encoded blocks. The results proves that pixels located closer to the intra prediction samples have higher correlation with them and therefore the weights should be higher closer to the reference samples.

1.1.4 Texture synthesis

In a paper from 2010 about *perceptual video coding: challenges and approaches*, by Zhenzhong et al the major challenges within the so called *perceptual video coding* and possible future research directions are discussed. The objective of perceptual video coding is to achieve maximum visual quality of the decoded video by taking the Human Visual System into consideration. [12]

The broad field of texture in-painting and synthesis has been well researched mostly from academic point of interest. Most of this research focuses on still images where a larger image is constructed from a small sample. In the last decade work has been done on extending this research to video as well, an example seen in the paper from Yu-Bei Lin and Xing-Ming Zhang [13]. Here they divide the area of *perceptual video coding* into five categories; Region-of-interest, saliency-map, Just Noticeable Difference, synthesis-based and hybrid methods. All these try to tackle the problem of reducing the bits needed to code the video but at the same time maintain a good visual result by investigating how the Human Visual System is perceiving certain textures. They all introduce quite complex and rigorous methods.

1.2 Aim of the thesis

The used sample based intra prediction in the current standard is efficient when it comes to simple structures such as edges and uniform regions, but not for more complex textures. The residual, i.e. the difference between the original and the predicted frame, becomes larger with worse predictions, and this then requires more bits to encode in the coding process. Therefore the intra prediction usually costs more bits to encode than the other variant of prediction, the inter prediction. Hence any improvements on the intra prediction could cause gains in the coding process.

Seen in section 1.1 Related work there have been previous studies that tests the block based intra prediction as well as a combination of the block based *inter* prediction with the sample based intra prediction. However, in this thesis the block based intra prediction is combined with the sample based intra prediction, and the sample based intra prediction in the latest standard H.265/HEVC is considered.

One motivation for the thesis is to determine the possibility of creating prediction tools that tries to improve the appearance of highly complex video content, such as water. Today temporal textures (e.g. water, leaves and smoke) are normally difficult to compress efficiently by video codecs due to their irregular spatial patterns and chaotic movement over time. This makes it interesting to explore alternate methods for compressing them.

Another background motivation to the thesis is to investigate the relatively unexplored area of combining video compression with techniques that are more focused towards perceptual similarity. Several fundamental questions exists related to how a video encoder, that focuses on being as pixel near to the original video as possible, can utilize methods for producing pixels which look similar for humans but likely not according to an objective metric like mean squared error. Works connecting to the texture synthesis area has been showed useful, as seen in section of related works 1.1.4, but these often involve complex algorithms and could be hard to implement. The goal in this thesis is however to try simple things first and see if a better perceptual result can be achieved by combining the known methods of sample based and block based intra prediction.

Two general problems that appear when compressing a sequence, especially for temporal dynamic structures, are *block artefacts* and *lack of details*. This thesis will try to concentrate on these problems as well as to reduce the error in the image and aim for the same visible quality over the whole image, which is important for the general impression. The focus has been on single frames in sequences.

Chapter 2

Background

2.1 Video compression overview

Video compression is about reducing and removing redundant information from the video data. Typically information from neighbouring pixels both within a picture but also from previous coded pictures are used to make a prediction of the video. Since the compression process is lossy, i.e. you loose information about the video sequence, the reconstructed video will always differ from the original video in some way. A major goal of any video codec standard is to provide tools to hide or minimize those distortions while still maintaining a high compression ratio to get the size of the video file as small as possible.

In other words, the core problem of video compression is to compress a video sequence without making the visual experience worse. To do this you would ideally remove parts that the Human Visual System are less sensitive to. This section briefly explains some concepts within video compression.

2.1.1 Structure of a video sequence

A video sequence is built up by consecutive still images, so called *frames*, that are presented after each other with a certain picture rate so that one gets the impression of motion. Each frame is built up by *pixels* (also called *samples*) forming a grid with each one having certain intensity, usually three different color components (see section 2.1.8 with color spaces). The number of pixels on the grid, the height and width, is the *resolution* of the image, for example the resolution 1920x1080 means that each frame is 1920 pixels wide and 1080 pixels high. An illustration of the structure of a video sequence can be seen in figure 2.1.

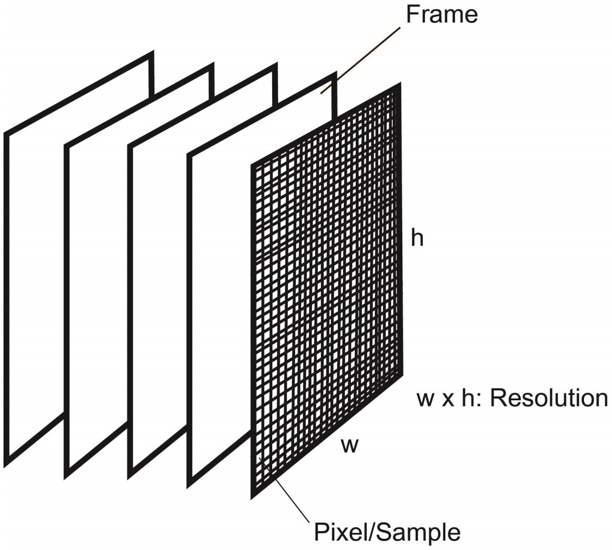
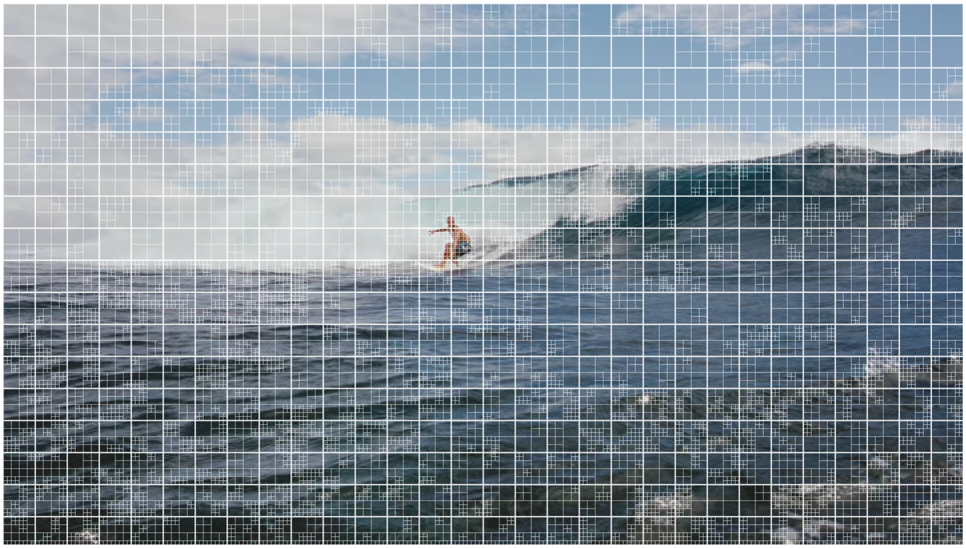


Figure 2.1: Illustration of a video sequence divided into frames. Each frame consists of a grid of pixels (also called samples) and the width times the height of each frame in pixels is the resolution of the video sequence.

2.1.2 Dividing a frame into blocks

When compressing a video sequence the frames are divided into non-overlapping blocks. Each of these blocks are predicted, transformed and coded into a bitstream in order to be able to transfer the video sequence to the decoder that later decompresses the sequence (see CODEC 2.1.3). The order in which the blocks in each frame is encoded is from the upper left corner and then row-wise through the whole frame. The division of each frame is first into Coding Tree Blocks (CTU) and these blocks can then be further divided into smaller Coding Units (CU). The largest coding unit is of size 64x64 (*Largest Coding Unit*, LCU). An integer number of CTB's makes up a *slice*, each picture consists of one or more slices, they are all *independently* decodable. The CU's are in parallel partitioned into a set of one or more Prediction Units (PU) and into one or more Transform Units (TU) [14]. In figure 2.2 the first frame from the sequence 'Surfing' is shown when divided into coding units and transform units. It can be seen that the blocks are smaller when the image contains more details.

For each CU a prediction mode (intra or inter, see section 2.1.3) is signalled to the bitstream [2]. The Prediction Units (that are of same or smaller size than the CU) can have the sizes 4x4, 8x8, 16x16, 32x32 or 64x64. The size is decided by testing several combinations of sizes of the block and choosing the one that gives the most efficient compression, which is the *Rate-Distortion optimization* (see section 2.1.6).



(a) Surfing, divided into Coding Units.

(b) Surfing, divided into Transform Units.

Figure 2.2: A frame from the sequence 'Surfing' divided into Coding Units and Transform Units. (Generated by the program Gitl HEVC Analyzer)

2.1.3 CODEC: encoder-decoder

The part of the algorithm that compresses a video is called *encoder*, and the one that reverses this operation to recreate the video is called *decoder*, together these make up a pair that is called a *CODEC*. Hence, the encoder takes an uncompressed video sequence as input and gives a compressed bitstream as output. It consists of several steps:

- **Prediction**:
  In order to reuse data to remove redundant information a prediction of each prediction unit is done.

  - **Intra**; is used to remove correlation within the same frame.
    The intra prediction uses previously coded pixels within the same frame when predicting a block. (See section 2.2 for more description)

  - **Inter**; is used to remove correlation between frames.
    The inter prediction uses the temporal dependency between frames to predict a block. The reference frames can either be past or future frames, depending on how the coding structure is built up. This method is frequently used within a coding sequence when lots of preceding frames already have been coded and the current frame can reuse lots of data from these. This makes this method cost less when it comes to the actual bits.

  In figure 2.5 the structure of a video sequence is shown and the arrows illustrates how the intra and inter refers to other blocks within the frame or to blocks in another frame respectively.

- **Transform and quantization**:
  Subtracting the prediction from the original signal creates the *residual*, which consequently is the part that could not be predicted by the prediction method. The residual still contains information that can be further compressed, and this is done by applying a transformation; the *Discrete Cosine Transform* (DCT) (except for 4x4 intra coded blocks that uses the Discrete Sine Transform). The residual signal is therefore represented by a number of transform coefficients which are quantized by division with a quantization step size Q_{step}. This step size is derived from a QUantization Parameter QP (see 2.1.5) These are then coded into the bitstream. [14]

- **Entropy coding**:
  This stage maps all the calculated elements for each block into binary code and bit-stream representations. Depending on the type of the information that is being coded, for example if it is a high-level property or not, the design of the coding is being done differently. [14]

In figure 2.3 an overview of the coder-decoder process can be seen and in figure 2.4 a more detailed block diagram of the encoder in HEVC is shown.

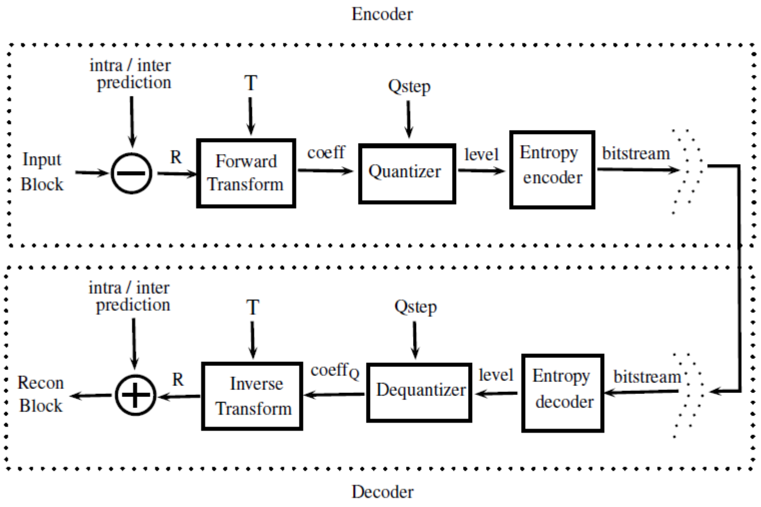


Figure 2.3: Overview of the block-based hybrid video coding in the HEVC standard. T is the tranform matrix, R is the residual, and Q_{step} is the quantization step size (from [15])

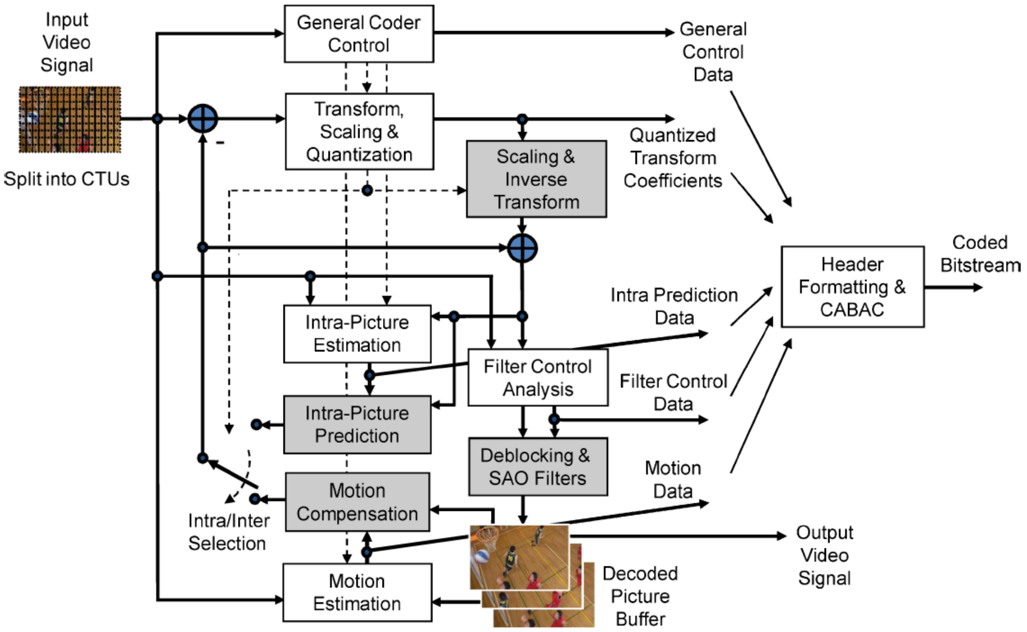


Figure 2.4: Block diagram of the HEVC encoder (with decoder modelling elements in gray). From *Overview of the High Efficiency Video Coding (HEVC) Standard* [1]. The CTU's are Coding Tree Units that the frame is divided into. The data generated by the prediction along with the chose mode and quantized coefficients as well as some other data are sent to the coding process CABAC where it is coded into a bitstream.

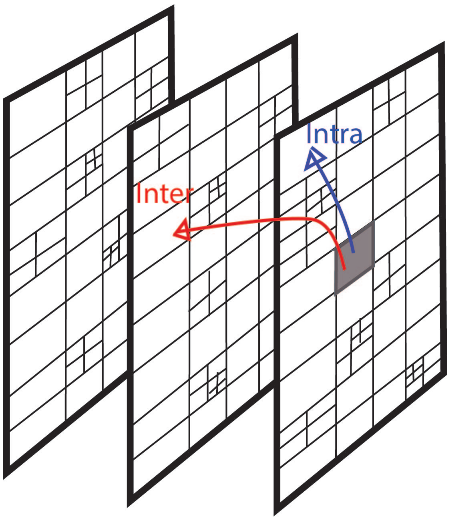


Figure 2.5: Sequence divided into frames with arrows representing the intra prediction using already coded blocks in the same frame and inter prediction using blocks in other frames.

2.1.4 Current standard: HEVC/H.265

High Efficiency Video Coding (HEVC) or H.265 is the latest standard and was released during 2013. It doubles the compression capability in comparison to its predecessor AVC/H.264 (*Advanced Video Coding*) without sacrificing quality. A standard within video coding specifies the decoder and so also the input to the decoder, however the encoder is not specified. The intra prediction in HEVC provides a higher number of angular modes (33 compared to 8 in AVC) which gives higher accuracy when predicting directional structures such as edges and lines [2].

2.1.5 Quantization parameter, QP

The Quantization Parameter QP is deciding the size of the quantization step size and hence the amount of quantization and the trade-off between bit-rate and quality. The larger QP-value the more loss of quality. The QP-parameter can be between 0 and 51. [14]

2.1.6 Rate-Distortion Optimization

The encoder in the video compression process has to decide which prediction mode and what prediction parameters to use for each block (see sample based intra prediction 2.2.1). This is an optimization between the used bitrate and the observed distortion that comes with this choice of mode and parameter; the *Rate-Distortion Optimization* [14], which can be simplified to minimize the cost J in:

$$J = \lambda * R + D, \tag{2.1}$$

where R is the bitrate and D the distortion. Lambda is a lagrangian multiplier.

2.1.7 In-loop Filtering

The in-loop filtering is located within the loop of encoding and decoding, and its purpose is to enhance the quality of the picture that is being displayed. This filter also affects the reference pixels for predicting the next blocks since it is located within the loop. Therefore it has a strong impact on the performance of the video coding scheme [14].

HEVC specifies two in-loop filters; a *deblocking filter* and a *sample adaptive offset* (SAO). The deblocking filter is applied first and deals with block boundary structures by operating over the prediction and transform block boundaries. The SAO-filter is applied to the output of the deblocking filter and corrects different levels that have been shifted due to quantization and also reduces ringing artefacts [14], [2].

2.1.8 Color space: YCbCr

To represent the color and light intensities of an image one uses a certain color space. The most common one in display systems is a mixture of red, green and blue, the RGB color space. For the coding and transmission the YCbCr color space is more established; *luminance* Y together with two *chrominance* components Cb and Cr. The chrominance components are representing the color impression, Cb for blue-difference and Cr for red-difference. The luminance (or luma) component represents the light-intensity of the image. Since the Human Visual System (see section 2.3) is less sensitive to color than to texture and structure information, the luma component is more important to have in high definition than the chroma components. Usually the chroma components are sub-sampled meaning that the samples of choma components are more sparse in the image. This is done to reduce the data needed to store the image. The notation of one the most commonly used sub-sampling is 4:2:0, the first number indicates luma samples and the two others represents the chroma samples relative to the luma. The second value specifies the horizontal sub-sampling and the last one being zero means that the same sub-sampling in vertical direction is the same as the horizontal. So this particular sub-sampling has half the resolution for Cb and Cr compared to Y, in both horizontal and vertical direction. [14]

2.2 Intra prediction

The method of Intra prediction uses previously coded blocks when predicting a new block within the frame. This type of prediction is typically applied when no previous frame has been coded or when there is a scene or illumination change in the sequence so that the Inter prediction doesn't work. The Intra prediction method typically requires more bits in the coding process due to worse prediction than inter prediction. The method can be divided into two areas; **Sample based intra prediction** (SBIP) and **Block based intra prediction** (BBIP). Here it is referred to the sample based as the method known from the latest standards H.264 and HEVC. This is effective when it comes to imitate locally simple texture such as edges, lines and planar surfaces.

The block based method showed up during the standardization of the HEVC Range extension and it occurred in a draft of this standard but was later removed. Today it is a part of a draft of HEVC screen content coding where it has been aligned with inter block prediction. This method is proposed as a method to improve more complex structures.

Both of the methods are described in more detail here below.

2.2.1 Sample based intra prediction (SBIP)

The sample based intra prediction (here abbreviated to SBIP) in video coding uses reference pixels in a frame to predict blocks within the same frame. Since the order in which the blocks in each frame is encoded is from the upper left corner and then row-wise through the whole frame already encoded pixels in the frame will be to the upper left of the next block. Intra prediction takes this into consideration when using the pixels to the left and above the block to predict pixels within the block.

The intra mode coding is a trade-off between a better prediction and less signalled bits. An increased number of intra modes gives a better prediction, but this also reduces the efficiency of the intra mode coding. For the luma components the three most probable modes are derived and the best one of these chosen.

In HEVC the intra prediction consist of three steps: reference sample array construction, sample prediction and post-processing. It can be classified into two categories: Angular prediction methods and DC/planar prediction methods. See table 2.1. The first category is supposed to model structures with directional edges, and the second category estimates smooth image content. The prediction is performed at the selected transform block size, from 4x4 to 32x32 samples. ([2], page 91)

To prevent introduction of artificial edges the HEVC standard introduces steps where the reference samples as well as the generated prediction boundary samples for DC and directly vertical and horizontal mode can be smoothed [16].

Table 2.1: The intra prediction modes

Intra prediction mode	**Name**
0	Planar
1	DC
2...34	Angular (N), N=2...34

Reference sample construction

The reference samples that will be used to predict the block are located along the left and top border of the block, seen in figure 2.6. The top reference vector is extended to the right for angular modes that predicts from the upper right corner, and the left reference vector is extended below for angular modes that predicts from the lower left corned (if these samples are available) as shown in the figure.

Sample prediction

- *Angular Prediction*
  In HEVC there are a set of 33 angular prediction directions at 1/32 sample

accuracy (see figure 2.6). Both vertical and horizontal. In Table 2.2 the angular parameter A is listed for each angular mode, the parameter describes the angularity, how many 1/32 sample grid units each row of samples is displaced with respect to the previous row, for each mode. ([2], page 97-98.)

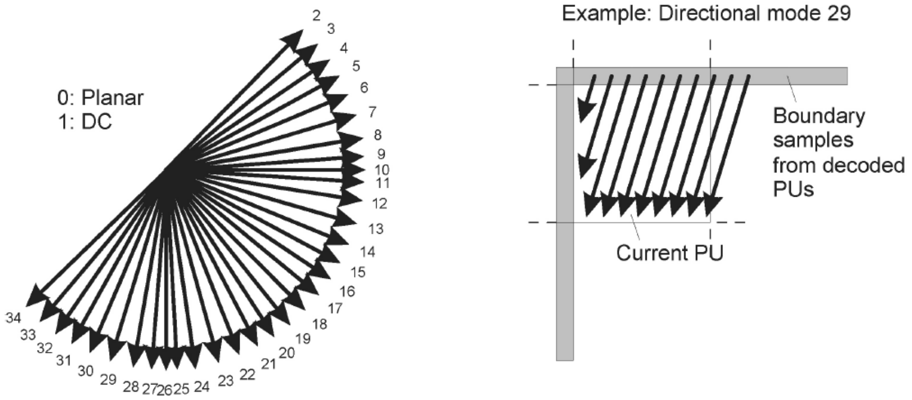


Figure 2.6: The intra modes and their directions of the prediction. From *Overview of High Efficiency Video Coding (HEVC)* by Wiegand et al [1].

Table 2.2: The angular parameter A for each angular mode. ([2], page 98)

Horizontal modes																	
Mode	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	
A	32	26	21	17	13	9	5	2	0	-2	-5	-9	-13	-17	-21	-26	
Vertical modes																	
Mode	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34
A	-32	-26	-21	-17	-13	-9	-5	-2	0	2	5	9	13	17	21	26	32

For the horizontal modes 2-10 and vertical modes 26-34 the reference samples will simply be the left reference sample row or the top reference row respectively:

$$ref[x] = p[-1+x][-1], (x >= 0) \tag{2.2}$$

for vertical modes and

$$ref[y] = p[-1][-1+y], (y >= 0) \tag{2.3}$$

for horizontal modes. ([2], page 99.)

For the rest of the angular modes, horizontal modes 11-17 and vertical modes 18-25, both reference samples from the left and top will be needed. The reference-arrays in equations 2.2 and 2.3 will be extended for negative indexes. For the vertical modes the top reference row is extended to the left by projecting samples from the left reference column, and for the horizontal modes the left reference column is extended upward by projecting samples from the top reference row. This is done through the equations:

$$ref[x] = p[-1][-1 + ((x * B + 128) >> 8)], (x < 0) \tag{2.4}$$

for vertical modes and

$$ref[y] = p[-1 + ((y * B + 128) >> 8)][-1], (y < 0) \tag{2.5}$$

for horizontal modes. $>>$ means a bitwise right shift ($>> 8$ means simply a division by 2^8). ([2], page 99.)

The parameter B is the inverse angle of the angular parameter A and the conversion from A to B is described in Table 2.3.

Table 2.3: The inverse angle parameter B as a function of angular parameter A. ([2], page 99)

A	-32	-26	-21	-17	-13	-9	-5	-2
B	-256	-315	-390	-482	-630	-910	-1638	-4096

Finally, the angular prediction is obtained by projecting the reference samples by using the prediction angle to interpolate the samples within the block. This is described with the equations:
For horizontal modes (2-17)

$$p[x][y] = ((32 - f) * ref[y + i + 1] + f * ref[y + i + 2] + 16) >> 5 \tag{2.6}$$

and for vertical modes (18-34)

$$p[x][y] = ((32 - f) * ref[x + i + 1] + f * ref[x + i + 2] + 16) >> 5 \tag{2.7}$$

$>>$ means a bitwise right shift ($>> 5$ means simply a division by 2^5).
([2], page 100.)

- *DC Prediction*
  The DC mode is simply to take the average of the reference samples immediately above and to the left of the block. This mode is best for homogeneous areas.

- *Planar Prediction*
  The planar prediction is designed for smooth areas where the DC prediction can result in blocking artefacts and the angular predictions can create visible edge-artefacts. It is achieved by averaging a horizontal and vertical linear prediction with the equations [2]:

$$p[x][y] = (p_h[x][y] + p_v[x][y] + N) >> (log_2(N) + 1) \tag{2.8}$$

where the horizontal prediction is

$$p_h[x][y] = (N - 1 - x) * p[-1][y] + (x + 1) * p[N][-1] \tag{2.9}$$

and the vertical prediction is

$$p_v[x][y] = (N - 1 - y) * p[x][-1] + (y + 1) * p[-1][N]. \tag{2.10}$$

$>>$ means a bitwise right shift ($>>$ $(log_2(N)+1)$ means a division by $2^{log_2(N)+1}$).

Post-processing
Since some of the modes can generate discontinuities over the boundaries of the prediction blocks a filtering over or along the edge is performed after the prediction. This is especially DC and directly horizontal and vertical modes [2].

Signalling of the prediction mode
In HEVC the prediction for the most probable modes for the current block are calculated. To decide which one that is being used Rate-Distortion Optimization is performed (see 2.1.6) and the mode selection is being sent as side information to the decoder.

2.2.2 Block based intra prediction (BBIP)

The block based intra prediction (here abbreviated to BBIP) is based on the idea to predict a block by entirely reusing already encoded blocks in the frame. The most similar block in a specified search area next to the current block is found by comparing the blocks with some metric. Therefore the block is predicted as a displacement from already reconstructed blocks. The position from where the block is copied is saved as a block copy vector with the vertical and horizontal position. This is then coded and sent to the decoder so this knows from where it should copy. The method has often been referred to as *Intra Block Copy* (IntraBC). It removes redundancy from repeating patterns which typically occur in text and graphic regions, i.e. screen content, and therefore it has been mostly tested for this type of content. The BBIP is inspired by a motion compensation method using a block matching technique in the Inter prediction method, but for the Inter method the copying is done from another image and not within the same [5]. Ballé J. and Wien M. showed in 2007 that depending on the content of the video sequence, substantial gains in rate-distortion performance (see 2.1.6) was achieved. Here they also referred to the method of using a template when looking for similar blocks to reuse within the frame, so called *template matching* [3]. As matching template an inverse-L area adjacent to the coding block could be used (for further description, see [9]). The template matching is applied so that the Intra Block Copy block vector, the vector pointing to the block being copied, can be derived from previously reconstructed regions on the decoder side without encoding in the bit stream. Because of the search for blocks involved in the block based intra method the time to encode increases, since typically a decoder is told what to do and not do any search [7]. The quality of the prediction could also be reduced since it is decided from samples outside the block and not the original inside the block as in IntraBC. To get better quality one could choose several candidates with the Template Matching and then compare these with the original block.

2.3 Human Visual System

The Human Visual System consists of two parts; the eyes and the brain. It is a complex and highly non-linear system which therefore is hard to model [17]. The sensitivity of the HVS depends on the background luminance and color of the stimuli and the visual attention is often object-based [12]. The Human Visual System is less sensitive to color than to texture and structure information. Because of its complexity, it is important to be aware of that there is a difference between the image that is displayed and the image that is actually perceived.

2.4 Video and Image quality assessment

There are two types of image and video evaluation; *subjective* and *objective*. With subjective evaluation being the subjective impression of image quality this is the most reliable way of assessing, because humans are the receivers of the image information being displayed. MOS, the mean opinion score is a subjective quality measurement obtained from some human observers, has been regarded as the most reliable form of quality measurement. Despite this, the MOS method is too slow, inconvenient and expensive for most applications [18]. Hence, it is important to find an objective quality metric, that is being calculated through an equation, that reflects the MOS. The goal of any quality assessment method is to measure the quality of images and videos in a perceptually consistent manner and in close agreement with subjective human judgements.

The objective quality metric can be used for evaluation or optimization in the process of finding the image of best quality. The focus is here on full-reference image quality metrics, meaning that a complete reference image is assumed to be known and used in the calculation of the metric.

The objective metrics that has been considered in this work are described below. VIF and SSIM has been developed to fit the HVS better than MSE and PSNR, although the later ones being the most frequently used.

2.4.1 Mean Squared Error

The Mean Squared Error between two images is defined as the average of the square of the errors (pixel differences) between the images ([19], page 281):

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(P_i - Q_i)^2 \tag{2.11}$$

where P_i denotes the pixels in the original image, Q_i the pixels in the reconstructed image and n denotes the total number of pixels. This metric is assuming that the loss of perceptual quality is directly related to the visibility if the error signal, and it is simple to calculate. Despite this, two distorted images with the same MSE may have very different type of errors [17]. the value of MSE can be very high although the blocks are perceptually similar (M Bosch et al, Segmentation-Based Video Compression Using Texture and Motion Models). [20]

2.4.2 Peak Signal-to-Noise Ratio

Peak Signal-to-Noise Ratio (PSNR) is very commonly used in image and video quality assessment. PSNR measures the relation of the mean squared error between the original image and the reconstructed image to the maximum value of the original image ([14], page 65):

$$PSNR = 10 * log_{10}(\frac{MAX_{orig}^2}{MSE}) \tag{2.12}$$

The fact that it is mathematically convenient in the context of optimization makes the PSNR frequently used [17]. However, PSNR has only a limited approximate relationship to the Human Visual System which makes it questionable in the sense of quality measure. In this thesis only the MSE is used (see section 2.4.1), but the PSNR is here mentioned since it is common within video quality assessment as a measure of distortion and it relates to the MSE.

2.4.3 Visual Information Fidelity

In a paper from Sheikh and Bovik, *A Visual Information Fidelity approach to Video Quality Assessment* ([21]), a visual information fidelity (VIF) approach is proposed to measure video quality. In the proposed information-fidelity framework the *mutual information* is used to measure how natural an image is. The mutual information of two variables is the measure of the mutual dependence of the variables. Sheikh and Bovik define stochastic models for the source, distortion and the Human Visual System in order to quantify the mutual information quantities. The visual information fidelity criterion is then derived from these. After some assumptions regarding the models they develop and implementation in Matlab, which is used in this work for evaluation. For a more detailed description of the Visual Information Fidelity and the implementation, see [21]. VIF is said to correlate more with how the Human Visual System perceives images, compared to the Mean Squared Error. The VIF-metric takes values between 0 and 1, where 1 is corresponding to a non-disturbed image.

2.4.4 Structural Similarity

In the paper *Image Quality Assessment: From Error Visibility to Structural Simlilarity* by Wang et. al. [17] a framework for quality assessment based on structural information is proposed. This assumes that the human visual perception is adapted to extract structural information from a scene and therefore a structural similarity (SSIM) index could replace a time-consuming subjective evaluation. According to the paper the natural image signals are highly structured and their pixels exhibit strong dependencies, especially those close to each other. The proposed method tries to take this into consideration when comparing structures instead of pointwise pixel difference. They define the structural information in an image as those attributes that represent the structure of objects in the scene, independent of the average luminance and contrast. For image quality assessment it is useful to apply the SSIM index locally rather than globally, the image is downsampled, and the result for all samples can then be averaged for the whole image to get a single overall quality measure; mean SSIM (MSSIM). A Matlab implementation of the MSSIM is provided from [17] which has been used in this work for evaluation. The SSIM-metric takes values between 0 and 1, where 1 is corresponding to a non-disturbed image.

Chapter 3

Method

3.1 Overview

The method was based on the Intra prediction method from the latest standard HEVC [2]. Subsequently a method of BBIP, Intra Block Copying, was introduced since it could improve the lack of details in the prediction blocks. These methods were combined in a way so that the error decreased and the visual appearance was improved.

Since the development was done continuously, with the purpose to maximize the visual improvement and minimize the error, it was an iterative working process and in this section the resulting steps from this method is presented.

The methods were constructed so that they were applied on an image blockwise. The blocksizes N=32,8 were chosen in order to see how the methods behaved for different blocksizes. The blocks were separated with some distance x_step, y_step so that one could observe how well the resulting block fitted into the neighbourhood and see if blocking artefacts appeared, see figure 3.1. The steps were put to from $2 * N$ to $4 * N$ for block size $N = 32$ and $4 * N$ to $8 * N$ for block size $N = 8$, depending on resolution and how many blocks that fitted into the image. The already compressed image, with chosen QP-value, was used as reference for the prediction and the original image was used as reference for calculating the error.

Some weight matrices were used to add the sample based and block based blocks together, shown in figure 3.2.

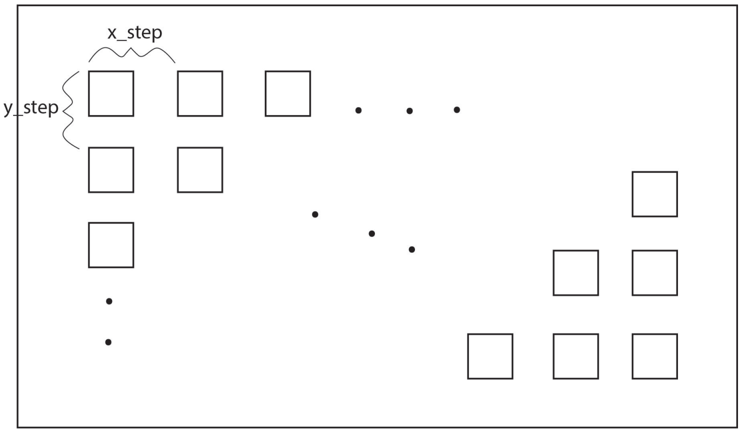


Figure 3.1: The method was applied on the image blockwise, with some distance *x_step* and *y_step*.

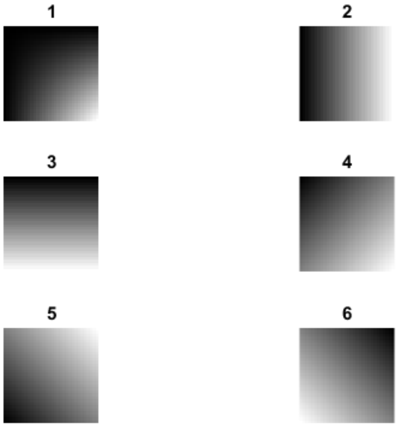


Figure 3.2: All of the weight matrices used when combining the SBIP block and the BBIP block, for block size N=32. wm_n n=1..6. Black color corresponds to 0 and white to 1. These matrices are used for the block based and the $(1 - wm_n)$ are used for the sample based intra prediction, so the total of the two weights always sum up to 1.

Since the luminance component is more important for the Human Visual System than the color components (see section 2.3), the Cb and Cr components were removed before applying the methods, and hence only black-and-white images were used. (See section 2.1.8 for a description of the YCbCr color space.)

The images used were not deblocked or SAO-filtered (see section 2.1.7) since the proposed method is supposed to be applied before these in the coding process.

3.2 Starting point; sample based intra prediction (SBIP)

First of all the Intra prediction scheme was implemented in Matlab according to the latest standard HEVC/H.265, section 2.2.1 taken from the book by Budagavi et. al. [2]. The most suitable Intra prediction mode was found with MSE (see section 2.4.1), although in the real case this is combined with optimizing against the coding cost, Rate-Distortion Optimization (see section 2.1.6), which has not been considered in this thesis. The Intra prediction is interpolated from pixels on the left and upper edge of the block and therefore one could expect it to be more accurate along these edges of the block and less accurate in the bottom right corner. The results were compared with the one from HM's implementation, an encoder for HEVC, to verify and see if the predictions were the same. The algorithm was applied on several images to see how it behaves for different content.

3.3 Investigating block based intra prediction (BBIP)

The idea was to improve the sample based prediction with a block based intra method, here we refer to the Intra Block Copy (IntraBC) method (see [4]). As mentioned in section 2.2.2, the IntraBC method is similar to the Inter prediction, but the Inter prediction often works quite badly for a temporal dynamic structure as water because of the irregular movements over time. Since water often has a periodical appearance *within* the same frame, the IntraBC method could be an alternative. By reusing blocks this method is also making sure the quality will not differ too much within the image. The goal of using block based intra prediction in this thesis was to see if already encoded blocks in an image can be used for future blocks in the same image, and if the visual result was sufficiently good.

The search range for each block was defined by three variables; rl-left range, rt-top range, and rr-upper right range, this is illustrated in figure 3.3 and 3.4. Although the search range is defined in whole numbers of blocks, the search performed at integer pixel positions within the search range (not only at block positions). To optimize and find the most suitable block, the Mean Squared Error was used.

The search range for Intra Block Copy was investigated in order to see how it affects the visual appearance and error.

3.4 Combined sample based and block based intra prediction

The next step, after investigating the SBIP and BBIP separately, was to combine these methods to see if this could yield a better result than the implemented version of sample based intra prediction. The sample based intra block ($SBIP$) was weighted together with the block based intra block ($BBIP$), in a way that the error decreased and the visual appearance was improved. For this some weights matrices were used ($wm_n, n = 1..6$), shown in figure 3.2. Samples closer to the reference samples in intra prediction is likely more reliable and the weight matrices were designed to vary the importance of the sample based intra prediction and the intra block copy in some of

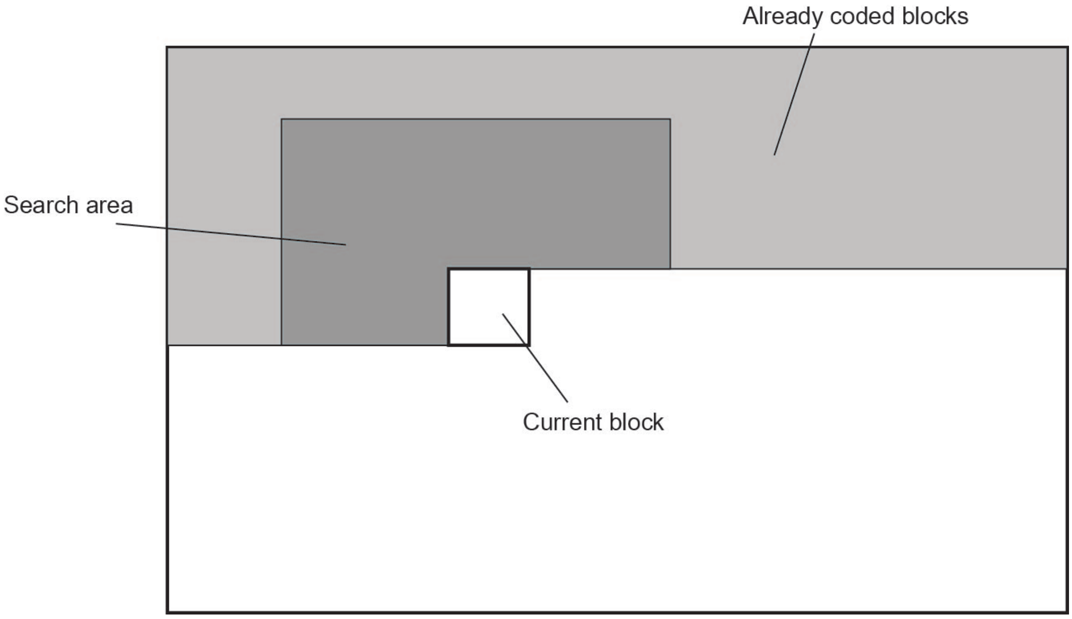


Figure 3.3: block based intra prediction range in image

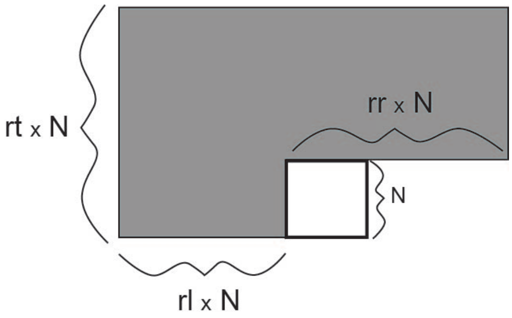


Figure 3.4: How the block based intra prediction range is defined with parameters rl, rt and rr.

the directions that is used by intra prediction. Each sample in the weight-matrices varies between 0 and 1, depending on how much of the sample based or block based that was considered. The sum of the weight for SBIP and weight for BBIP always adds up to 1.

For all the block based intra predictions, the **search range** $rl = 3, rt = 3, rr = 3$ was used. The procedure to investigate the combination of the blocks was as following:

3.4.1 Average

The simplest combination: each block of Intra prediction and Intra Block Copy was found independently (each by choosing the one with least MSE) and the result was put to be the average of these for each block position, the **independent version**:

$$predictedBlock(x,y) = \frac{SBIP(x,y) + BBIP(x,y)}{2}, \tag{3.1}$$

where $SBIP(x,y)$ is the sample based intra block and $BBIP(x,y)$ is the block based intra block at position (x,y) within the block.

Another way to add the blocks is to first find the sample based intra as usual (with MSE) and then, for each block position in the search range, take the average between this BBIP block together with the SBIP block. Choose the one that gives the least MSE of the total block, the **dependent version**:

- Calculate *SBIP* by choosing the mode that gives least MSE
- Take the average of the blocks as in equation 3.1.
- Choose the *BBIP* that gives the least MSE of *predictedBlock*.

3.4.2 Weight matrix 1

The sample based intra prediction block and Intra Block Copy block was found separately and weighted together with a weight matrix that took the sample based intra prediction along the upper end left edge and gradually go over to Intra Block Copy prediction in the lower right corner (Weight matrix 1, see Figure 3.2). The equation was as following, for each position (x,y) within the block:

$$predictedBlock(x,y) = (1-wm_1(x,y))*SBIP(x,y)+wm_1(x,y)*BBIP(x,y), \tag{3.2}$$

where $SBIP(x,y)$ is the sample based intra block, $BBIP(x,y)$ is the block based intra block and $wm_1(x,y)$ is the weight matrix 1 all at position (x,y) within the block. The separate blocks were found by choosing the one with the least MSE.

3.4.3 Directional weight matrices, matrices 2-6

The sample based intra prediction block and Intra Block Copy block was found separately and combined by applying one out of five different directional weight matrices (see Figure 3.2, matrices 2-6). This was tested in order to investigate if an increased number of weight matrices could improve the performance. The separate blocks were found by choosing the one with the least MSE and the weight matrix for a block was chosen by testing to combine the blocks with each matrix and then choosing the one with least MSE.

The procedure was as following:

- For all n=2..6, calculate for each position (x,y) within the block:

  $$predictedBlock(x,y)_n = (1-wm_n(x,y))*SBIP(x,y)+wm_n(x,y)*BBIP(x,y), \tag{3.3}$$

where wm_n is the weight matrix n, $SBIP$ is the sample based intra block and $BBIP(x, y$ is the block based intra block.

- Choose the $predictedBlock_n$ that gives the least MSE.

This method increases the overhead information since the decoder need to know which of the weight matrices being used for each block.

It was also investigated whether it exist a correlation between the angle of the mode (see figure 2.6) and the angle of the weight matrix (see figure 3.2). The sample based intra prediction is interpolating in the block with certain angle that makes it better along the edges from where it is interpolated. Therefore a naive thought would be that when optimizing against MSE, also the weight matrices are chosen so that the sample based intra prediction is preserved at these edges where it predicts at its best. This was investigated by looking at each weight matrix and see which modes were chosen for this one.

3.4.4 Horizontal and vertical weight matrices, (weight matrix 2 and 3)

A horizontal weight matrix (wm_2) was used for horizontal intra modes and DC and planar modes (modes 0-17) and a vertical weight matrix (wm_3) was used for vertical intra modes (modes 18-34) (see Figure 2.6 and 3.2). The weight matrices are hence chosen implicitly, depending on which intra prediction mode is used. The reason to test this was that it would be interesting to see if the extra signalled bits when using matrices 2-6 (due to signalling of which matrix that is chosen) could be reduced without loosing too much of the error-reduction obtained there. The horizontal and vertical weight matrices were chosen since the sample based intra is supposed to predict at its best along the edges, and especially the edges from where the directional intra modes are predicted (see section 2.2) so these matrices would be the best ones for the horizontal and vertical directions. An increased amount of weight matrices could also be interesting to test, but in lack of time this was outside the scope for this thesis.

The merging of the blocks was done in two ways:

- ***Independent search***: sample based intra prediction block and Intra Block Copy block was found separately with MSE and combined with appropriate weight matrix.

  For horizontal modes, DC and planar modes, for each position (x,y) within the block:

  $$predictedBlock(x, y) = (1 - wm_2(x, y)) * SBIP(x, y) + wm_2(x, y) * BBIP(x, y), \tag{3.4}$$

  and for vertical modes, for each position (x,y) within the block:

  $$predictedBlock(x, y) = (1 - wm_3(x, y)) * SBIP(x, y) + wm_3(x, y) * BBIP(x, y), \tag{3.5}$$

  where $wm_2(x, y)$ and $wm_3(x, y)$ are the weights for coordinates (x,y) within the block, $SBIP$ is the sample based intra block and $BBIP$ is the block based intra block (which are found for each block position).

- ***Dependent search***: The Intra Block Copy block was found depending on which *combined* block that gave the least MSE. Hence the Intra Block Copying is dependent on Intra prediction and weight matrix. So the procedure was as following:

  - Choosing the sample based intra block (intra mode) as usual by taking the one that gives the least MSE.
  - For all block based intra blocks in the search range:
    add the sample based and block based blocks together for each (x,y) within the block with appropriate weight matrix as follows.

    For horizontal modes, DC and planar modes, for each position (x,y) within the block:

    $$predictedBlock(x,y) = (1-wm_2(x,y)) * SBIP(x,y) + wm_2(x,y) * BBIP(x,y), \tag{3.6}$$

    and for vertical modes, for each position (x,y) within the block:

    $$predictedBlock(x,y) = (1-wm_3(x,y)) * SBIP(x,y) + wm_3(x,y) * BBIP(x,y), \tag{3.7}$$

    where $wm_2(x,y)$ and $wm_3(x,y)$ are the weights at position (x,y) within the block, $SBIP$ is the sample based intra block and $BBIP$ is the block based intra block. The MSE for each of the total blocks was calculated.
  - Choose the total block that gives least MSE.

3.4.5 Horizontal and vertical matrices - dependent; Exhaustive search

In the dependent search as in above: look at *all* combinations of Intra prediction mode and Intra Block Copy block and choose the one with least MSE of the total block. Hence an *exhaustive search.* So all the intra modes together with all block based intra blocks were tested.

3.4.6 Refined Template Matching method combined with sample based intra prediction

The Template Matching method was tested as a substitute to the Intra Block Copy method in combining with the sample based intra block since it is a way to reduce the bits needed by avoid sending the position data for the copying (which is used in Intra Block Copy) [9]. Although this reduces the data needed, the complexity at the decoder side is increased since it needs to derive the position. Also the quality could be reduced since the prediction is decided from samples outside the block and not the original sample values. The template matching uses a L-shaped template along the left and upper borders of the block when finding the best matching block, see figure 3.5. It matches this template with all possible L-shaped areas in the search-range, instead of matching the actual block. This is because the decoder does not have the information that isn't already encoded. A "refinement method" of Template Matching is here used when combining it with the sample based intra block (as in

[9]): The K best blocks with Template Matching were found and then each of these were combined with the sample based in order to choose the one that gives the least MSE of the combination. The method has to signal which of the "best blocks" that is used, but this does not require as many bits as sending the block vector in IntraBC.

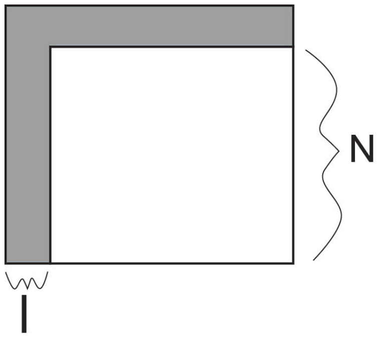


Figure 3.5: The template in the Template Matching method is defined as the pixels to the left and on top of the current block, with the pixel width l.

For each template matching block k = 1-K:
horizontal modes, DC and planar modes, for each position (x,y) within the block:

$$predictedBlock(x,y) = (1-wm_2(x,y))*SBIP(x,y)+wm_2(x,y)*templateMatching_K(k)(x,y), \tag{3.8}$$

vertical modes, for each position (x,y) within the block:

$$predictedBlock(x,y) = (1-wm_3(x,y))*SBIP(x,y)+wm_3(x,y)*templateMatching_K(k)(x,y), \tag{3.9}$$

where $wm_2(x,y)$ and $wm_3(x,y)$ are the weights at position (x,y) within the block, $SBIP$ is the sample based intra block and $templateMatching_K(k)$ is the k'th of the K best template matching block.
The combination that gave the least MSE was chosen.

The parameters were put to:

- Template width was put to l=N/4 in order to capture some structure but not make it too large (This results in l=8 for blocksize N=32 and l=2 for blocksize N=8).
- The search range was defined the same way as for Intra Block Copy.
- The K=10 best blocks was used in the refinement method.

3.5 Evaluation

The result from the methods was compared to the sample based intra prediction method in H.265/HEVC in order to see if it could give any improvements to the current standard.

As this thesis is focusing on water sequences, five sequences containing water was chosen for the testing, seen in table 3.1. The sequences 'DropsOnWater' and 'CalmingWater' are from a testset released last year (VIL: BVI_Textures, published 30 Jan 2015 by Miltiadis Alexios Papadopoulos). The sequence 'DucksTakeOff' is from the SVT (Swedish Television) High Definition Multi Format Test Set (Feb 2006). The sequence 'Riverbed' is from Taurus Media Technik (recorded 2001).The *exhaustive* method was also tested on some standard sequences to see a more general result for sequences not just containing water. All the images were generated with deblocking- and SAO-filter turned off (see section In-loop filtering 2.1.7), since intra prediction typically is applied to a decompressed image before in-loop filtering.

For each sequence only the first frame of each sequence was used. The five frames from the test set containing water are shown in figure 3.6.

All sequences used had the YCrCb-colorspace with sample format 4:2:0 (described in section 2.1.8). Only the luma-component was used.

Table 3.1: Sequences in the water test set and their resolution.

Sequence	**Resolution**
DucksTakeOff ("Ducks")	1280 × 720
DropsOnWater ("Drops")	1920 × 1080
Surfing	1920 × 1080
Riverbed	1920 × 1080
CalmingWater	1920 × 1080

The methods were tested with the parameters:

- **Quantization parameter**, QP: $22, 27, 32, 37$
- **Block size**, N: 32×32, 8×8 pixels

in order to investigate the behaviour for different quantization and block size.

In lack of any other good and robust objective metric, the Mean Squared Error was used in the methods when optimizing for best blocks and also the mean of the MSE for all the blocks in an image was used when comparing the methods, even though this is likely not suitable since the MSE does not very well correlate with the Human Visual System (see section 2.4.1). In order to further evaluate the method and to use metrics that correlate better with the HVS, the Visual Information Fidelity (VIF) and Mean Structural Similarity (MSSIM) (section 2.4) were also calculated for the total resulting image with the blocks.

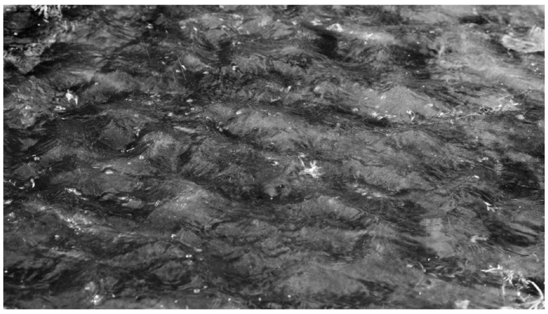
Moreover a comparison along the right edge was introduced, to try to see if this correlates with the blocking artefacts, since the intra prediction could produce bad result at the right edge since it interpolates from the left and above. The right edge was chosen since most blocking artefacts was observed here and this edge seemed to

(a) 'Drops'.

(b) 'Ducks take off'.

(c) 'Riverbed'.

(d) 'Surfing'. (Only the part of the image with water is used)



(e) 'Calming water'.

Figure 3.6: Images from the water test set used in the evaluation. All of them are the first frame in each sequence. Only the luma component is used (black and white images).

improve the most when combining the sample based and block based predictions, but it hasn't been fully investigated, so a more extensive edge-metric should look at all the edges around the block in order to measure block artefacts. The proposed *edge-metric* here compares the right edge of the resulting block with the neighbourhood in the image and compare this difference with the difference in the original image as follows:

$$edgeMetric = \frac{1}{N} \sum (d_{block} - d_{originalBlock})^2, \tag{3.10}$$

where N is the block size and d is the pointwise difference between the right edge of the block and the column to its right (in the background image);

$$d = e1 - e2, \tag{3.11}$$

with $e1$ and $e2$ being the edge and the column (see figure 3.7). Thus, the metric is the Mean Squared Error between the edge-difference in the resulting image and the edge-difference in the original image. The value for each block was calculated and, in the same way as for the MSE, averaged for all blocks in the image to get a quality measure for each image. This edge-metric was used for the horizontal and vertical weight matrices (see section 3.4).

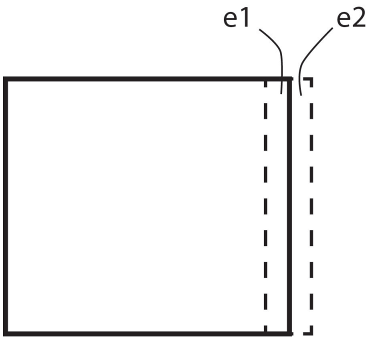


Figure 3.7: The proposed edge metric compares the right edge of the block with the background directly to the right, and then compares this value with the one for the original.

Chapter 4

Results & Discussion

The results from the method are showed in tables and figures below. The results in the tables are in **percentage difference from the sample based intra prediction**. For mean MSE a *negative* percentage is reduction of error whereas for SSIM and VIF it is a *positive* percentage. Since the VIF and SSIM has another order of magnitude than the MSE (VIF and MSSIM are between 0 and 1 and the mean MSE was typically over 200 for these tests) the resulting outcome in %-difference from sample based intra can also be smaller since the variation is not as large as for MSE. Hence, it is more interesting to observe whether the VIF and MSSIM-results were *positive* or *negative*, as an indication on improvements from sample based intra or not.

For the blocksize N=8 it was harder to spot the difference visually, both in block artefacts and details, due to the small blocks, so the visual results are mostly based on the appearance of the N=32 blocks.

4.1 Search range for BBIP (IntraBC)

As can be seen in figures 4.1 and 4.2, the mean MSE for all the blocks in an image decreases with increasing search range, for both block size N=32 and N=8. For simplicity the left, top and right search ranges were put to the same value (rl=rt=rr) and the mean MSE for all blocks is calculated for each value from range 1 to 8 with steps of 0.5 blocks. Increasing the search range is therefore a way to make the error decrease, but the time consumption and computational complexity is increasing with increasing range, so the search range should not be higher than necessary. Consequently, it is important to find parameters for the search range that gives a sufficiently good result for both error and complexity. Observing the figures it can be seen that after a certain range the change in error is not as large as for lower search ranges. After this observation the search range is put to $rl = rt = rr = 3$ since this seems to be a sufficiently good search range for both block sizes. In table 4.1 the values for the mean MSE over all blocks in an image when applying the sample based intra prediction on the water test set can be seen. This can be compared with the values in the plots for the block based intra prediction which shows that the SBIP has lower mean MSE than the BBIP.

Table 4.1: The mean MSE for the **sample based intra prediction (SBIP)** applied on the water test set sequences when using QP=37.

mean MSE QP=37	N=32	N=8
Ducks	711	151
Drops	226	56
Surfing	131	31
Riverbed	230	48
CalmingWater	216	54

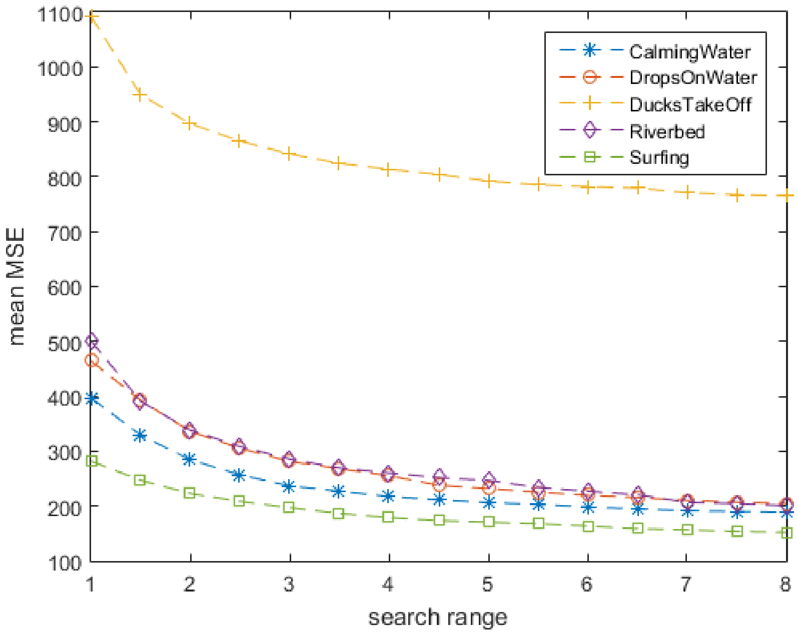


Figure 4.1: Mean MSE for varying search range on IntraBC. N=32, QP=37 (same trends for other QP-values.)

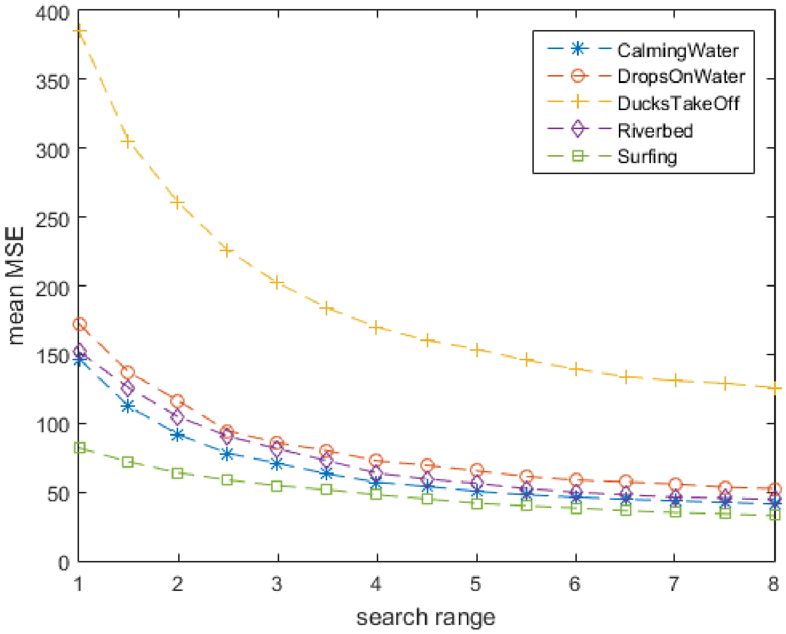
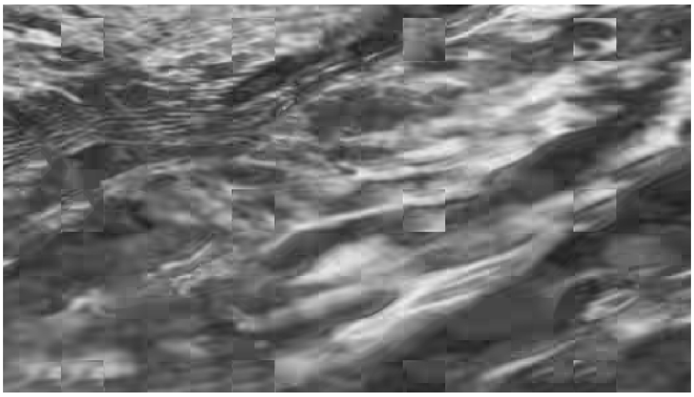
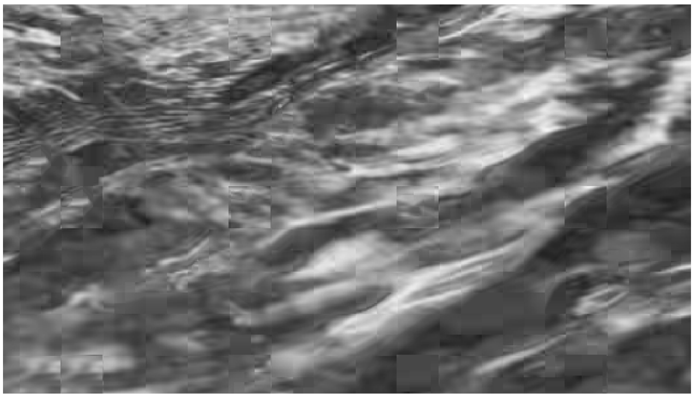


Figure 4.2: Mean MSE for varying search range (rl=rt=rr) on IntraBC. N=32, QP=37 (same trends for other QP-values.)

In figure 4.3 a part of the frame from 'Riverbed' is shown when using intra block copy with ranges 1,3 and 8. Here it can be seen that the blocks are most visible for search range 1, and fits better into the background when the range is 3 or 8.



(a) search range rl=rt=rr=1



(b) search range rl=rt=rr=3

(c) search range rl=rt=rr=8

Figure 4.3: Intra Block Copy on a part of 'Riverbed', QP=37, N=32. For different search ranges.

4.2 Average

The results from testing the independent and dependent versions of the average on the water test set is shown in table 4.2 for blocksize N=32, and in table 4.3 for blocksize N=8.

For almost all cases the average-methods give error reduction compared to sample based intra, the reduction in mean MSE ranges from -5% to -30. 'Surfing' at blocksize N=8 for the independent average is the only that gets worse mean MSE than sample based intra; approximately 5%. For the MSSIM and VIF results, the percentage difference from sample based intra is small, but almost for every sequence and QP it is a positive percentage which means it could be an improvement from the sample based intra. Most gains in VIF and MSSIM are observed for 'Ducks' with the MSSIM-metric for N=32 and with the VIF-metric for N=8. It can be seen that for all the sequences the dependent version gives more gains than the independent version.

All the percentages for MSSIM at blocksize N=8 are very low. An explanation could be that since the blocks are separated with some distance this evaluation metric might not capture the change in structure when the blocks are too small.

With increasing QP-value, the error-reduction seem to increase for blocksize N=32 and decrease for blocksize N=8. This is more discussed when comparing the methods in section 4.8.2.

In figure 4.4 some visual results are shown from sequences 'Ducks' and 'CalmingWater' when applying the independent average and the dependent average as well as only the sample based intra. For 'CalmingWater' it is difficult to distinguish the blocks since the variance is high, although at some places the sample based intra blocks are visible. For 'Ducks' the structure is more regular and the blocks can be distinguished. The sample based intra is being the one with most blocking artefacts, and the average-methods seem to add more structure to the blocks as well as to improve the right edge of the blocks where the sample based intra is more inaccurate in its prediction. This is observed for several sequences. However, for some blocks the accurate left edge for sample based intra is not preserved when taking the average with the block based. The visual results of the independent and the dependent method are very similar so it is hard to tell which of them that are visually better by just looking, but the numerical results discussed above indicate on the dependent method being better.

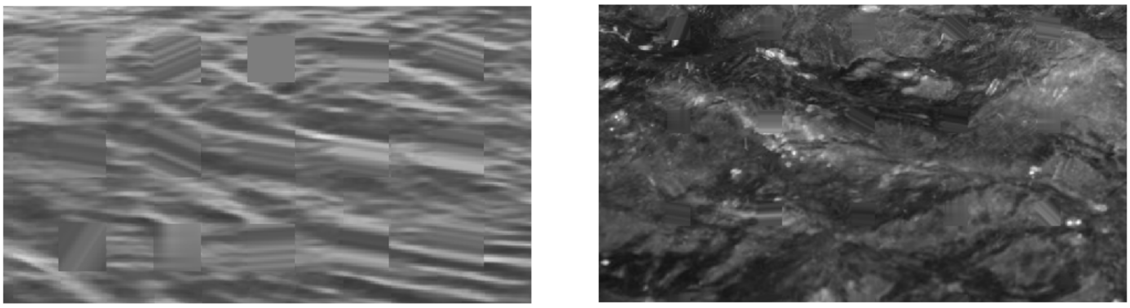
The results from the first combinations of SBIP and BBIP indicates on that a combination could add more details in the image, as well as reduce the MSE error and also that searching the block based intra dependently of the sample based intra seem to improve the gains.

Table 4.2: Average between the sample based intra and block based intra (IntraBC), the blocks found either *independently* or *dependently*. Results shown in percentage difference from sample based intra. rl=rt=rr=3, N=32.

%		**Average**					
N=32		**Independent**			**Dependent**		
	QP	mean MSE	VIF	MSSIM	mean MSE	VIF	MSSIM
Ducks	22	-19.91	-0.15	1.60	-25.73	0.40	2.19
	27	-20.05	0.06	1.63	-25.81	0.80	2.22
	32	-19.57	0.47	1.64	-25.78	1.28	2.29
	37	-20.24	1.16	1.76	-26.54	2.09	2.41
Drops	22	-13.51	0.10	0.25	-19.05	0.19	0.36
	27	-13.85	0.11	0.25	-19.03	0.20	0.37
	32	-13.76	0.20	0.24	-20.18	0.34	0.39
	37	-14.86	0.38	0.31	-20.15	0.53	0.43
Surfing	22	-9.46	-0.51	0.22	-18.46	-0.01	0.46
	27	-10.92	-0.22	0.26	-18.6	0.37	0.48
	32	-11.81	0.32	0.29	-19.39	0.83	0.50
	37	-13.74	0.90	0.36	-21.61	1.66	0.60
Riverbed	22	-15.53	0.05	0.30	-19.92	0.18	0.43
	27	-15.67	0.11	0.29	-20.03	0.24	0.42
	32	-15.44	0.24	0.29	-21.30	0.43	0.45
	37	-16.46	0.34	0.31	-22.71	0.63	0.50
CalmingWater	22	-18.84	0.18	0.36	-25.07	0.35	0.51
	27	-18.75	0.25	0.36	-25.07	0.43	0.50
	32	-19.05	0.34	0.36	-25.31	0.56	0.50
	37	-20.28	0.55	0.40	-26.41	0.75	0.55

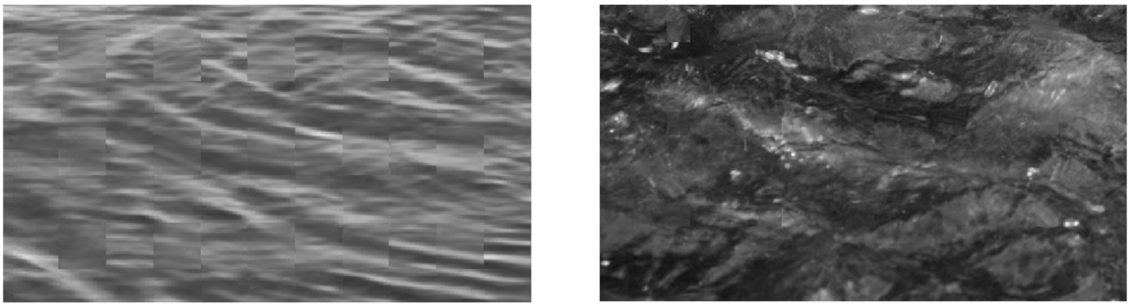
Table 4.3: Average between the sample based intra and block based intra (IntraBC), the blocks found either *independently* or *dependently*. Results shown in percentage difference from sample based intra. rl=rt=rr=3, N=8.

% **N=8**		**Average**					
		Independent			**Dependent**		
	QP	mean MSE	VIF	MSSIM	mean MSE	VIF	MSSIM
Ducks	22	-19.86	2.00	0.12	-29.38	3.58	0.18
	27	-19.67	1.99	0.12	-29.14	3.38	0.17
	32	-19.12	1.77	0.12	-28.43	2.96	0.18
	37	-19.37	1.53	0.13	-29.36	2.50	0.20
Drops	22	-10.78	0.39	0.03	-18.48	0.67	0.04
	27	-10.65	0.35	0.03	-18.85	0.61	0.04
	32	-11.99	0.36	0.03	-19.46	0.56	0.04
	37	-10.31	0.23	0.03	-17.50	0.42	0.04
Surfing	22	5.78	0.08	0.01	-4.00	0.77	0.03
	27	4.86	0.11	0.02	-4.52	0.72	0.03
	32	6.65	0.04	0.01	-2.26	0.54	0.03
	37	4.90	0.14	0.01	-3.75	0.59	0.02
Riverbed	22	-8.32	0.14	0.01	-17.34	0.41	0.02
	27	-8.75	0.16	0.01	-17.73	0.41	0.02
	32	-8.90	0.15	0.01	-17.31	0.36	0.02
	37	-5.06	0.09	0.01	-14.54	0.29	0.02
CalmingWater	22	-16.57	0.45	0.03	-23.37	0.71	0.04
	27	-16.69	0.41	0.03	-23.44	0.64	0.04
	32	-14.52	0.30	0.03	-22.29	0.53	0.04
	37	-14.53	0.27	0.03	-21.24	0.46	0.04



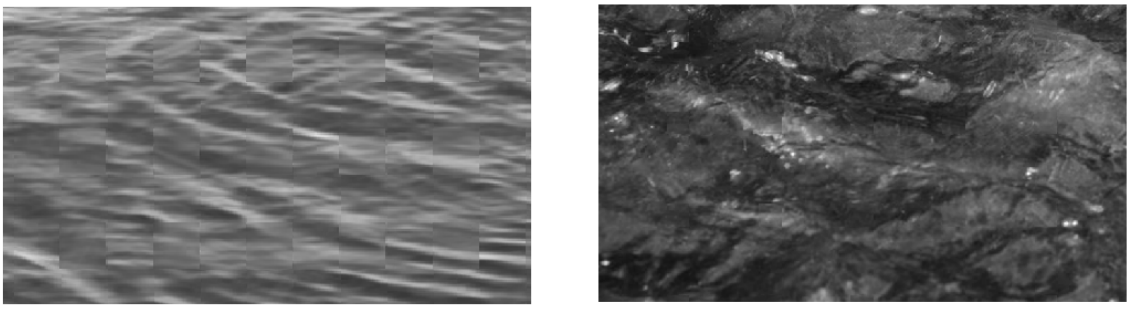
(a) Sample based intra prediction

(b) Sample based intra prediction



(c) Sample based and block based intra prediction combined with average, independently.

(d) Sample based and block based intra prediction combined with average, independently.



(e) Sample based and block based intra prediction combined with average, dependently.

(f) Sample based and block based intra prediction combined with average, dependently.

Figure 4.4: From 'Ducks' (left) and 'CalmingWater' (right), QP=27, N=32, rl=rt=rr=3.

4.3 Weight matrix 1, weight matrices 2-6

The results from applying weight matrix 1 together with the results when applying weight matrices 2-6 depending on which gives least MSE is shown in table 4.4 for blocksize N=32 and 4.5 for blocksize N=8.

For all the tested sequences the weight matrices 2-6 gives more MSE reduction than using matrix 1. For matrix 1 the mean MSE reduction ranges from -10% to -18% for blocksize N=32 and -7% to -16% for blocksize N=8 (except 'Surfing' which has around 0% for blocksize N=8). For matrices 2-6 the mean MSE reduction is from -17% to -24% for blocksize N=32 and from -18% to -29% for blocksize N=8 (again except 'Surfing', with the lowest error reduction at around -7%).

The VIF and MSSIM-values are all positive for both methods which could indicate an improvement in visual quality. The improvements in percentage from sample based intra are generally higher for matrices 2-6 than for matrix 1. The sequence 'Ducks' has the highest improvements from sample based intra; for blocksize N=32 the VIF and MSSIM is improved with approximately 1.3% for weight matrix 1 and $1-2\%$ for the directional weight matrices, and for block size N=8 this sequence shows some improvements in VIF; 1.5% for matrix 1 and $2.3-3.5\%$ for matrices 2-6. But like the results from the average-method, the MSSIM-gains are all very low for block size N=8, so it might be questionable if the gains maybe could be neglected for smaller block sizes.

For most sequences when using blocksize N=32 there is more reduction in error for higher QP. But the opposite can be observed for blocksize N=8. This was also observed for the average methods. This might be due to the distortion that is increased for higher QP-values, which could affect the smaller blocks more and the error reduction becomes smaller. This is more discussed in section 4.8.2 when comparing all the methods.

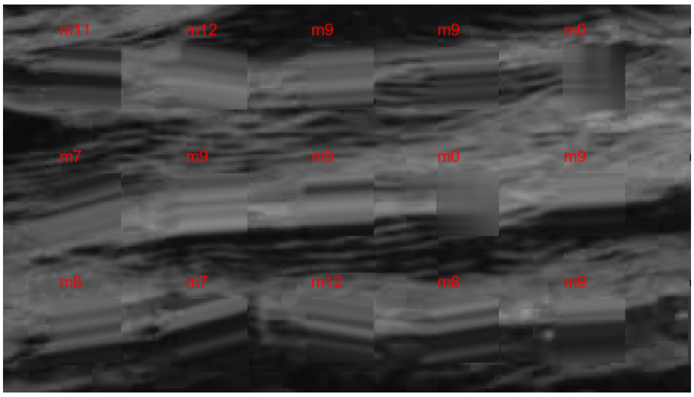
A cropped image from the test sequence 'Surfing' can also be seen in figure 4.5. The sample based intra is shown together with the combination of sample based and block based (IntraBC) with matrix 1 or matrices 2-6. The resulting intra modes and weight matrices are marked out (f for the weight matrix and m for the mode). Here it can bee seen that the combination of SBIP and BBIP seems to add more details in the blocks compared to sample based intra, especially for modes 0 and 1 (planar and DC mode). Also a reduction of block-artefacts could be noticed, most frequent when using weight matrices 2-6. However, for most blocks the sample based intra prediction gives good predictions at left and upper edges, and the block based intra prediction does not seem to be needed here. This motivates the use of weight matrices that uses only sample based intra prediction along the edges from where it is predicted.

Table 4.4: Results when applying weight matrix 1 or weight matrices 2-6 (depending on which gives least MSE). Difference from sample based intra in percent. rl=rt=rr=3, N=32.

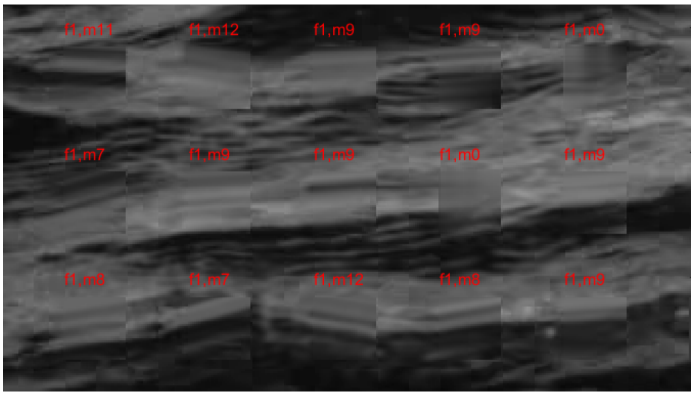
%		**Weight matrices**					
N=32		**1**			**2-6**		
	QP	mean MSE	VIF	MSSIM	mean MSE	VIF	MSSIM
Ducks	22	-18.16	1.13	1.34	-24.29	1.13	2.03
	27	-18.37	1.29	1.39	-24.93	1.45	2.10
	32	-17.48	1.37	1.34	-24.38	1.77	2.10
	37	-17.22	1.62	1.37	-24.30	2.30	2.18
Drops	22	-11.03	0.16	0.20	-18.17	0.22	0.34
	27	-10.97	0.15	0.20	-18.32	0.21	0.33
	32	-11.17	0.21	0.19	-18.15	0.33	0.32
	37	-12.35	0.30	0.25	-20.09	0.57	0.43
Surfing	22	-12.85	0.52	0.30	-16.62	0.41	0.41
	27	-13.54	0.66	0.32	-17.77	0.76	0.46
	32	-13.53	0.80	0.32	-18.27	1.18	0.49
	37	-13.41	0.99	0.33	-20.58	1.81	0.57
Riverbed	22	-14.83	0.24	0.27	-20.89	0.26	0.39
	27	-14.59	0.27	0.26	-21.2	0.35	0.4
	32	-15.08	0.35	0.28	-21.15	0.48	0.41
	37	-15.45	0.39	0.28	-22.06	0.61	0.43
CalmingWater	22	-14.63	0.27	0.29	-23.74	0.38	0.48
	27	-14.49	0.30	0.29	-23.44	0.42	0.48
	32	-14.77	0.35	0.29	-24.08	0.55	0.48
	37	-15.69	0.48	0.33	-24.73	0.76	0.52

Table 4.5: Results when applying weight matrix 1 or weight matrices 2-6 (depending on which gives least MSE). Difference from sample based intra in percent. rl=rt=rr=3, N=8.

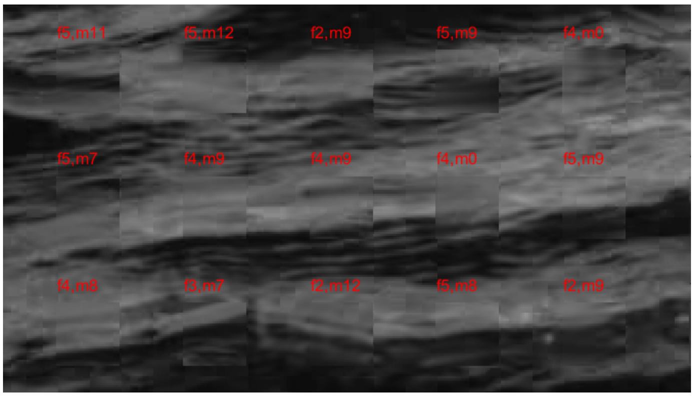
% **N=8**		**Weight matrices** **1**			**2-6**		
	QP	mean MSE	VIF	MSSIM	mean MSE	VIF	MSSIM
Ducks	22	-16.40	1.70	0.09	-29.29	3.48	0.17
	27	-16.83	1.66	0.09	-29.62	3.38	0.17
	32	-15.80	1.45	0.09	-28.82	2.84	0.17
	37	-16.12	1.19	0.10	-28.75	2.34	0.18
Drops	22	-10.92	0.27	0.02	-22.74	0.64	0.04
	27	-10.19	0.26	0.02	-22.28	0.59	0.04
	32	-11.75	0.26	0.02	-23.61	0.57	0.04
	37	-10.50	0.18	0.02	-21.85	0.41	0.04
Surfing	22	0.47	0.27	0.01	-7.77	0.76	0.03
	27	-0.05	0.26	0.02	-8.20	0.73	0.03
	32	1.74	0.15	0.01	-7.04	0.57	0.03
	37	2.71	0.07	0.01	-6.54	0.55	0.03
Riverbed	22	-13.39	0.21	0.01	-22.42	0.42	0.03
	27	-12.87	0.20	0.01	-23.14	0.44	0.03
	32	-12.48	0.17	0.01	-23.37	0.40	0.03
	37	-7.04	0.08	0.01	-18.64	0.29	0.03
CalmingWater	22	-11.52	0.3	0.02	-26.51	0.72	0.05
	27	-12.52	0.29	0.02	-26.06	0.64	0.05
	32	-11.13	0.21	0.02	-24.37	0.50	0.04
	37	-10.32	0.18	0.02	-23.62	0.44	0.04



(a) Sample based intra prediction



(b) Sample based and block based (IntraBC) intra prediction combined with weight matrix 1



(c) Sample based and block based (IntraBC) intra prediction combined with weight matrices 2-6 (depending on which gives least MSE).

Figure 4.5: From 'Surfing', QP=27, N=32. The weight matrix is marked with an f and the intra mode is marked with an m.

4.4 Investigating correlation between intra mode and direction of weight matrix

Several weight matrices to choose from seems to reduce the MSE error more than just using a single one, at least when comparing the result from weight matrix 1 with the result from weight matrices 2-6, but the question is whether it is worth the additional complexity and time-consumption of searching through the weight matrices? Therefore it would be interesting to see if a correlation between the intra-mode direction and the direction of the weight matrix exist, in that case the weight matrix could be automatically put to the corresponding one after chosen intra-mode.

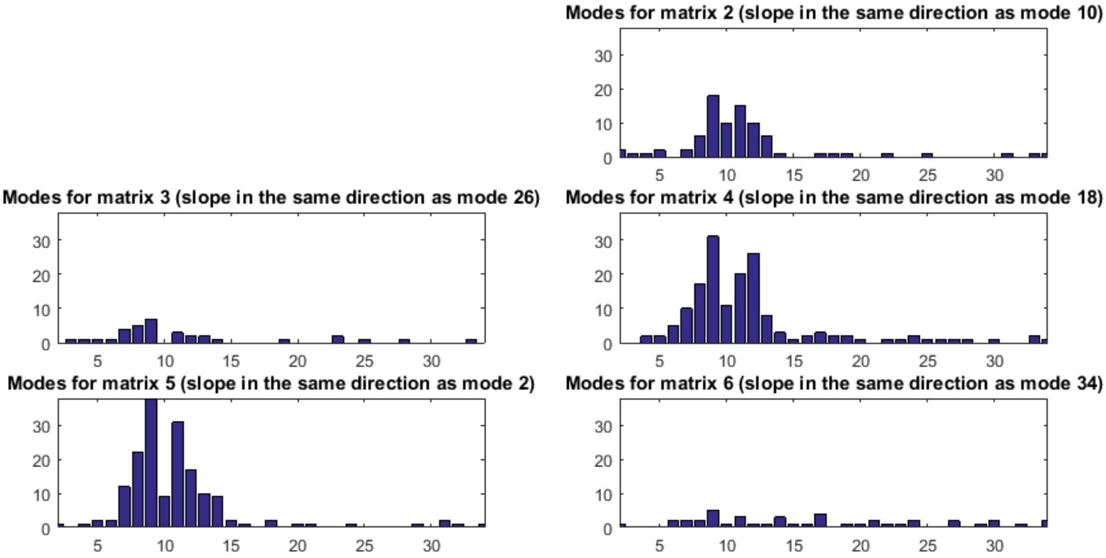
The resulting plots from testing the correlation between the intra mode and chosen weight matrix on the water test set are shown in figure 4.6 for blocksize N=32 and figure 4.7 for blocksize N=8, both with quantization parameter QP=22 (similar results was observed for other QP-values).

If a correlation between mode and weight matrix would exist one could hope to see this in the occurrence plot for each matrix. A direct connection between direction of mode and direction of the weight matrix would divide the modes into the following groups, with connected matrices:
Weight matrix 2 - Modes 7-13
Weight matrix 3 - Modes 23-29
Weight matrix 4 - Modes 14-22
Weight matrix 5 - Modes 2-6
Weight matrix 6 - Modes 30-34

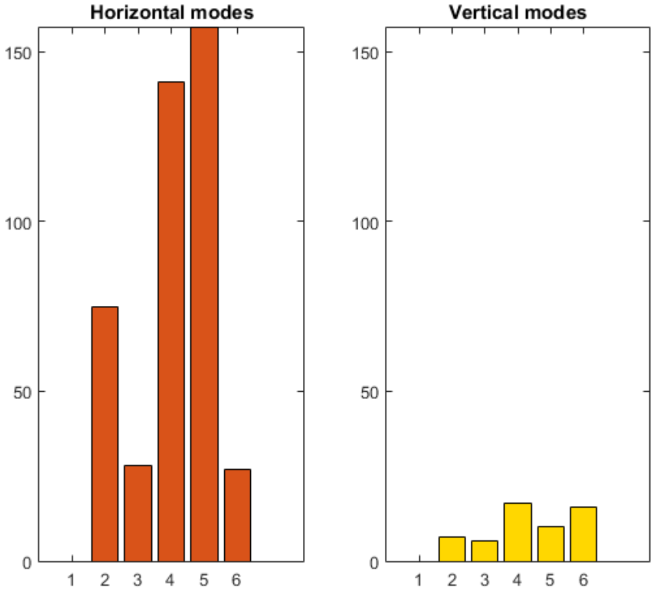
As can be seen in the figures this kind of relationship does not seem to exist; the modes seem to choose weight matrix quite random. There are a much larger number of horizontal modes which makes sense since the water surfaces are mostly horizontal.

It looks like there are many other aspects in the image that affects the decision of weight matrix, and not only the direction of the mode.

Still, the sample based intra prediction is more accurate along the edges and the decision of weight matrix could be connected to the direction in some way. In this thesis the horizontal and vertical weight matrix are chosen (for horizontal and vertical modes respectively) to test a lower number of weight matrices that are implicitly decided from the intra mode. These matrices are chosen since the sample based intra prediction is assumed to predict better at for samples closer to the reference samples.

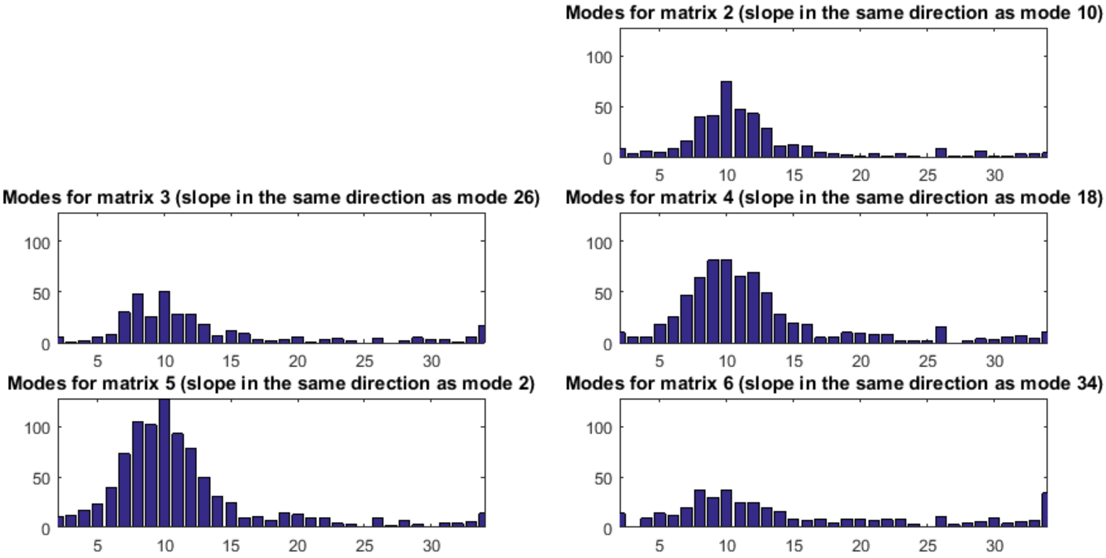


(a) Occurrence of the angular modes for each of the weight matrices 2-6.

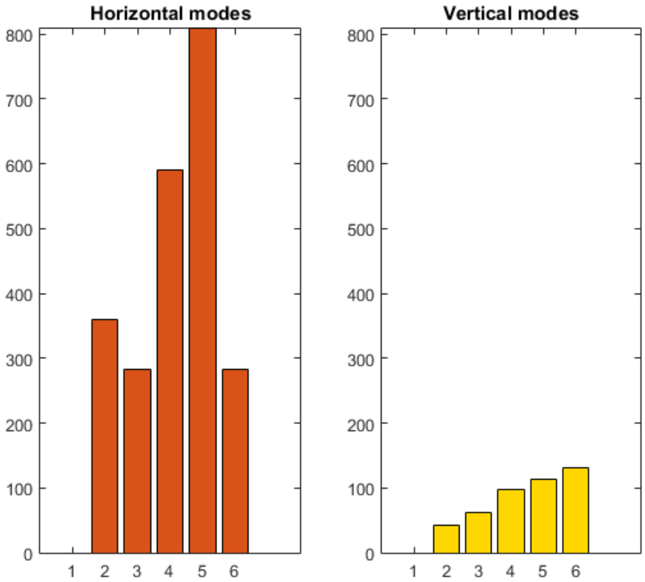


(b) The angular modes divided into horizontal and vertical modes, the occurrence of weight matrices for these two groups.

Figure 4.6: For all the sequences in the water test-set, N=32, QP=22 (similar results obtained for different QP). Notice that weight matrix 1 was not used in this test.



(a) Occurrence of the angular modes for each of the weight matrices 2-6.



(b) The angular modes divided into horizontal and vertical modes, and the occurrence of weight matrices for these two groups.

Figure 4.7: For all the sequences in the water test-set, N=8, QP=22 (similar results obtained for different QP). Notice that weight matrix 1 was not used in this test.

4.5 Horizontal and vertical weight matrices

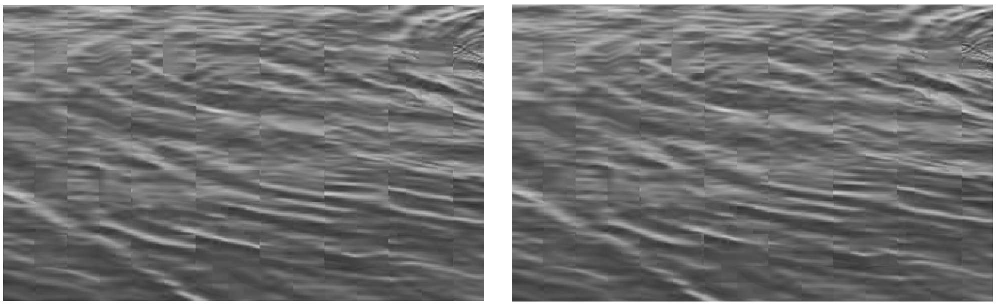
The results from the dependent and independent search when using weight matrix 2 and 3 (horizontal and vertical weigh matrices) can be seen in table 4.6 for blocksize N=32 and table 4.7 for blocksize N=8.

In table 4.6, for N=32, there can be seen that all the mean MSE for both independent and dependent has decreased compared to sample based intra. The dependent search reduces the mean MSE more than the independent search. The same trends can be observed in table 4.6, for blocksize N=8.

For the VIF and MSSIM, the percentage difference from sample based intra is positive for all test, which indicates an improvement. The dependent search has a larger improvement, especially for the sequence 'Ducks' with gains up to 2.72% for the MSSIM. The MSSIM-gain are very low for the smaller blocksize N=8, as observed earlier for both the average and matrix 1 as well as for matrices 2-6.

An example of visual result is shown in figure 4.8 which is a part from the results on test sequence 'Ducks', blocksize N=32, quantization parameter QP=37 and search range rl,rt,rr=3. Here it can be seen that the dependent search has slightly better edges and hence less blocking artefacts than the independent search which shows sharper edges. This was observed in several images.

The proposed edgeMetric seems to indicate an improvement for the dependent search compared to block based intra, but for the independent search the error has increased compared to intra. For 'Drops' with block size N=32 and for 'Surfing' with block size N=8 the dependent search also gives a higher edgeMetric than sample based intra with a lot of variation, so it is hard to draw any conclusions. But when looking at the visual results the metric seems to somewhat follow when the edges are getting better when using the dependent search instead of the independent, seen in figure 4.8. It could be an idea to involve the variance in the edge-metric since it seems to influence how visible the actual block artefacts are. It is also important to consider that this edgeMetric only estimates the right edge whereas the bottom edge also could be in consideration.



(a) Independent

(b) Dependent

Figure 4.8: A part from the frame from the sequence 'DucksTakeOff' with independent and dependent search when using horizontal and vertical weight matrices for the combination of sample based and block based intra. (N=32, QP=37 and rl,rt,rr=3)

Table 4.6: Difference from sample based intra, in percent. independent and dependent search. N=32. Horizontal weight matrix, matrix number two, for horizontal intra modes (as well as DC and planar) and vertical weight matrix, matrix number three , for vertical intra modes.

N=32		**Horizontal and vertical weight matrices**							
%		**Independent**				**Dependent**			
	QP	mean MSE	VIF	MSSIM	edgeMetric	mean MSE	VIF	MSSIM	edgeMetric
Ducks	22	-19.11	0.86	1.61	-3.82	-28.65	1.94	2.47	-23.18
	27	-20.00	1.15	1.70	-6.88	-29.28	2.3	2.57	-24.91
	32	-19.04	1.25	1.67	-1.35	-28.67	2.57	2.58	-23.66
	37	-18.82	1.70	1.70	-3.33	-29.12	3.27	2.72	-21.97
Drops	22	-10.08	0.03	0.18	23.00	-20.03	0.28	0.41	5.45
	27	-9.34	0.01	0.17	28.47	-19.79	0.33	0.41	2.06
	32	-10.04	0.12	0.16	18.44	-20.58	0.45	0.43	4.30
	37	-11.54	0.27	0.24	17.85	-20.89	0.58	0.47	0.79
Surfing	22	-7.38	0.13	0.21	20.33	-20.36	0.99	0.56	-5.93
	27	-8.18	0.34	0.24	18.39	-20.18	1.28	0.56	-3.67
	32	-8.66	0.59	0.26	18.13	-21.07	1.71	0.63	-9.54
	37	-10.98	0.97	0.31	14.17	-22.46	2.24	0.68	-11.94
Riverbed	22	-12.31	0.11	0.26	4.76	-21.13	0.40	0.48	-13.57
	27	-12.72	0.16	0.25	4.06	-21.38	0.49	0.46	-14.97
	32	-12.95	0.31	0.27	0.45	-22.66	0.65	0.50	-15.76
	37	-13.73	0.35	0.27	4.55	-23.23	0.77	0.53	-18.14
CalmingWater	22	-14.88	0.12	0.28	18.19	-25.86	0.53	0.57	-20.07
	27	-14.71	0.18	0.28	17.84	-25.87	0.57	0.56	-16.51
	32	-15.38	0.25	0.29	16.97	-26.18	0.69	0.56	-21.88
	37	-16.66	0.47	0.33	3.85	-26.60	0.89	0.57	-24.25

Table 4.7: Difference from Intra, in percent. independent and dependent search. N=8. Horizontal weight matrix, matrix number two, for horizontal intra modes (as well as DC and planar) and vertical weight matrix, matrix number three , for vertical intra modes.

N=8		**Horizontal and vertical weight matrices**							
%		**Independent**				**Dependent**			
	QP	mean MSE	VIF	MSSIM	edgeMetric	mean MSE	VIF	MSSIM	edgeMetric
Ducks	22	-14.91	1.76	0.10	3.18	-31.23	4.00	0.18	-31.1
	27	-15.29	1.82	0.10	1.71	-31.68	3.84	0.18	-30.15
	32	-14.63	1.58	0.10	0.86	-31.05	3.30	0.19	-28.63
	37	-14.83	1.42	0.12	2.46	-30.56	2.61	0.20	-24.12
Drops	22	-5.11	0.22	0.02	21.97	-23.29	0.73	0.04	-18.07
	27	-4.57	0.21	0.02	22.29	-22.80	0.66	0.04	-16.93
	32	-6.78	0.24	0.02	15.28	-23.30	0.58	0.04	-18.54
	37	-4.59	0.15	0.02	16.04	-21.85	0.45	0.04	-16.40
Surfing	22	14.11	-0.11	0.01	40.68	-4.66	0.85	0.03	0.19
	27	12.50	-0.05	0.01	40.92	-5.31	0.80	0.04	2.42
	32	14.42	-0.12	0.01	37.30	-3.01	0.55	0.03	5.72
	37	14.30	-0.07	0.00	35.45	-1.05	0.52	0.02	14.01
Riverbed	22	-2.49	0.08	0.01	27.21	-20.63	0.53	0.03	-7.38
	27	-2.51	0.09	0.01	26.98	-20.28	0.50	0.03	-10.47
	32	-2.75	0.10	0.01	25.91	-20.80	0.45	0.03	-8.46
	37	3.11	0.01	0.01	31.30	-18.61	0.34	0.03	-8.26
CalmingWater	22	-8.22	0.27	0.02	9.56	-25.37	0.75	0.04	-26.10
	27	-9.42	0.27	0.02	7.29	-25.68	0.67	0.04	-26.42
	32	-7.75	0.19	0.02	12.87	-24.74	0.54	0.04	-23.08
	37	-7.55	0.18	0.02	3.56	-23.00	0.44	0.04	-26.83

4.6 Exhaustive search

In table 4.8 and 4.9 the results from the exhaustive search is shown together with the dependent search using horizontal and vertical weight matrices, for block sizes N=32 and N=8. It can be seen that the reduction in error is larger for the exhaustive search than for the dependent search, which was expected since the exhaustive searches through all possible intra modes and IntraBC-block combinations and therefore should find a total block with less error than the dependent which is only searching for the best intraBC-block for a single intra-mode.

The VIF and MSSIM also seems to have some improvements for most sequences, and all of them better than dependent.

In Appendix the results from the exhaustive search for 17 standard sequences with different resolution is shown, in percentage difference from sample based intra and with block size N=32. For all the sequences the exhaustive search gives a reduction in MSE, ranging from approximately -14% to -41%. These results indicates on that the exhaustive method seems to work also for different content than just water.

The best results observed is for the sequence 'PeopleOnStreet', which is a sequence with a lot of people seen from a distance. This could also be considered as a complex texture.

In figure 4.9 the images from sample based intra prediction and exhaustive search on 'BasketballDrillText' are shown. The blocks covering the floor seems to give better results with the combined method since there are more details and also less blocking effects can be seen.

Table 4.8: Exhaustive search and dependent search when using horizontal weight matrix, matrix 2, for horizontal intra modes (as well as DC and planar) and vertical weigh matrix, matrix 3, for vertical intra modes. Difference from sample based intra prediction in percent. Blocksize N=32.

N=32		**Horizontal and vertical weight matrices**					
%		**Exhaustive**			**Dependent**		
	QP	mean MSE	VIF	MSSIM	mean MSE	VIF	MSSIM
Ducks	22	-33.93	2.96	3.02	-28.65	1.94	2.47
	27	-34.23	3.29	3.03	-29.28	2.30	2.57
	32	-34.32	3.77	3.18	-28.67	2.57	2.58
	37	-34.70	4.48	3.30	-29.12	3.27	2.72
Drops	22	-29.99	0.58	0.64	-20.03	0.28	0.41
	27	-29.99	0.65	0.64	-19.79	0.33	0.41
	32	-30.63	0.85	0.68	-20.58	0.45	0.43
	37	-30.87	1.06	0.73	-20.89	0.58	0.47
Surfing	22	-27.56	1.69	0.79	-20.36	0.99	0.56
	27	-27.56	2.10	0.79	-20.18	1.28	0.56
	32	-28.47	2.56	0.85	-21.07	1.71	0.63
	37	-29.82	3.2	0.92	-22.46	2.24	0.68
Riverbed	22	-32.47	0.74	0.67	-21.13	0.40	0.48
	27	-32.71	0.82	0.64	-21.38	0.49	0.46
	32	-32.96	1.01	0.67	-22.66	0.65	0.5
	37	-33.2	1.17	0.7	-23.23	0.77	0.53
CalmingWater	22	-33.15	0.86	0.8	-25.86	0.53	0.57
	27	-32.93	0.94	0.79	-25.87	0.57	0.56
	32	-33.16	1.07	0.78	-26.18	0.69	0.56
	37	-33.79	1.30	0.82	-26.60	0.89	0.57

Table 4.9: Exhaustive search and dependent search when using horizontal weight matrix, matrix 2, for horizontal intra modes (as well as DC and planar) and vertical weigh matrix, matrix 3, for vertical intra modes. Difference from sample based intra prediction in percent. Blocksize N=8.

N=8		**Horizontal and vertical weight matrices**					
%		**Exhaustive**			**Dependent**		
	QP	mean MSE	VIF	MSSIM	mean MSE	VIF	MSSIM
Ducks	22	-41.89	5.88	0.24	-31.23	4.00	0.18
	27	-42.09	5.49	0.24	-31.68	3.84	0.18
	32	-41.42	4.66	0.25	-31.05	3.3	0.19
	37	-40.74	3.71	0.26	-30.56	2.61	0.20
Drops	22	-44.03	1.38	0.07	-23.29	0.73	0.04
	27	-44.77	1.26	0.07	-22.80	0.66	0.04
	32	-43.95	1.09	0.07	-23.30	0.58	0.04
	37	-42.18	0.86	0.07	-21.85	0.45	0.04
Surfing	22	-23.16	1.97	0.06	-4.66	0.85	0.03
	27	-23.19	1.81	0.06	-5.31	0.80	0.04
	32	-20.25	1.37	0.05	-3.01	0.55	0.03
	37	-18.43	1.25	0.05	-1.05	0.52	0.02
Riverbed	22	-41.82	1.06	0.05	-20.63	0.53	0.03
	27	-41.76	0.99	0.05	-20.28	0.50	0.03
	32	-41.25	0.86	0.05	-20.80	0.45	0.03
	37	-38.17	0.69	0.05	-18.61	0.34	0.03
CalmingWater	22	-42.83	1.34	0.06	-25.37	0.75	0.04
	27	-42.67	1.18	0.06	-25.68	0.67	0.04
	32	-41.82	0.99	0.06	-24.74	0.54	0.04
	37	-40.11	0.82	0.06	-23.00	0.44	0.04

(a) Sample based intra prediction.

(b) Sample based and block based intra prediction combined in the exhaustive search.

Figure 4.9: 'BasketballDrillText', N=32, rl=rt=rr=3, QP=22.

4.7 Template Matching combined with sample based intra prediction

In table 4.10 and 4.11 the results is shown when evaluated with mean MSE, VIF and MSSIM compared to sample based intra, for blocksize N=32 and N=8 respectively. Here it can be seen that the error for the template matching method together with sample based intra has increased compared to the sample based intra. With blocksize N=8 the results are worse than for blocksize N=32.

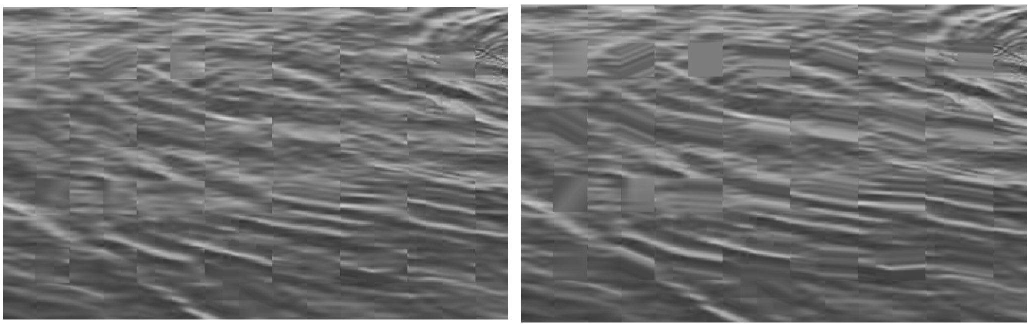
In figure 4.10 the Template Matching combined with sample based intra prediction (left) together with only the sample based intra prediction (right) is shown for a part of the sequence 'Ducks'. The template matching method has template width l=8 and is using the 10 best blocks from the template matching to find the best block together with sample based intra. The quantization parameter is QP=37 and the range $rl = rt = rr = 3$. One can observe that the Template Matching seems to add some more detail into the blocks but it also makes some of the blocking artefacts increase.

So the Template Matching does not seem to reduce the error compared to sample based intra as the IntraBC did in previous tests. But this can also be due to parameter setting; maybe the template width l should be increased or the number of calculated "best candidates" K should be higher. And the search-range was not extended to reach the "same area" as the IntraBC, which could affect the method so that it doesn't find the same structures as in IntraBC.

Even though the Template Matching here gives worse results than the IntraBC-combination it is important to remember the rate-distortion optimization and the fact that a lower signalling cost (need to send less information to the decoder) is preferable. Also, only the average result for all blocks in an image is here considered, and the results blockwise might be different. So the Template Matching method could very well be suitable for the blocks that gives a good result. Looking at blockwise results war unfortunately out of the scope for this thesis.

Table 4.10: Template Matching combined with sample based intra, K=10 best blocks, template width l=8, blocksize N=32, search range rl=rt=rr=3.

N=32 %		**Template Matching**		
	QP	mean MSE	VIF	MSSIM
Ducks	22	22.39	-2.02	-1.81
	27	21.73	-2.05	-1.81
	32	21.94	-2.03	-1.78
	37	20.02	-1.80	-1.72
Drops	22	58.02	-0.69	-0.71
	27	57.69	-0.74	-0.73
	32	62.13	-0.78	-0.79
	37	60.79	-0.82	-0.77
Surfing	22	61.54	-2.51	-1.31
	27	61.21	-2.71	-1.34
	32	57.26	-2.70	-1.28
	37	53.70	-2.69	-1.23
Riverbed	22	54.06	-0.65	-0.66
	27	53.40	-0.66	-0.67
	32	53.75	-0.67	-0.68
	37	51.62	-0.70	-0.66
CalmingWater	22	42.43	-0.66	-0.73
	27	42.41	-0.64	-0.72
	32	45.05	-0.76	-0.78
	37	44.85	-0.66	-0.78



(a) Sample based intra combined with refinement method of Template Matching.

(b) Sample based intra prediction

.

Figure 4.10: A part from the frame from the sequence 'DucksTakeOff' with sample based intra prediction combined with Template matching (left) or just the sample based intra prediction (right). In the refinement method the K=10 best blocks were used and the template width l=8.

Table 4.11: Template Matching combined with sample based intra, K=10 best blocks, template width l=2 ,blocksize N=8, search range rl=rt=rr=3.

N=8 %		**Template Matching**		
	QP	mean MSE	VIF	MSSIM
Ducks	22	35.53	-2.80	-0.16
	27	36.23	-2.65	-0.16
	32	38.15	-2.31	-0.17
	37	35.76	-1.69	-0.15
Drops	22	86.13	-1.04	-0.10
	27	78.82	-0.92	-0.09
	32	81.89	-0.84	-0.11
	37	89.73	-0.83	-0.12
Surfing	22	65.51	-2.20	-0.09
	27	64.67	-1.99	-0.09
	32	70.42	-1.84	-0.11
	37	58.24	-1.30	-0.09
Riverbed	22	93.84	-1.24	-0.10
	27	95.98	-1.18	-0.10
	32	88.65	-0.99	-0.10
	37	89.97	-0.90	-0.10
CalmingWater	22	60.69	-0.93	-0.09
	27	62.48	-0.92	-0.10
	32	60.64	-0.80	-0.10
	37	64.61	-0.70	-0.11

4.8 Concluding discussion

4.8.1 Estimate signalling cost and computational complexity

The actual coding process, how to integrate the methods into the current standard, isn't covered in this thesis. However it is important to have in mind how the different methods would affect the coding when evaluating them, especially if they cost more or less bits to code. Here it is discussed for each method what this computational complexity and cost could be.

In the implementation of the sample based intra prediction in this thesis the intra modes takes only the MSE into consideration when choosing mode, and the actual intra prediction in HEVC takes the coding cost into consideration (see section 2.1.6 Rate-Distortion Optimization). Therefore the resulting modes of this implementation could be a bit different from the ones derived in HEVC, but the aim with this work is only to investigate the effects of combining the sample based with the block based and not the whole implementation process.

When looking at the algorithm for the rate-distortion optimization (equation 2.1) the choice of which mode is selected is based on the bitrate R and the distortion D. It is supposed that if any of the the models proposed in this thesis would be implemented in the actual coding process, it would be considered as an extra alternative in this optimization process, and hence the same equation of rate-distortion optimization is applied to it. From this equation it can be derived that improving the distortion with D_{red} allows the bitrate to increase with $R_{red} = D_{red}/\lambda$. So by improving the prediction, i.e. reducing the distortion, the bitrate could increase and it still would generate the same total cost $R * \lambda + D$.

The Rate-Distortion Optimization is done blockwise, and so the combined method of sample based and block based intra prediction cold be chosen only when it gives better results than the sample based intra prediction in the standard HEVC. In this thesis the average over all blocks in an image was considered, when applying the same method for all blocks, but in a real implementation the algorithms are only considered for each block.

For all the proposed models that are based on the Intra Block Copy technique of BBIP an additional signalling of the block vector (pointing at the block being copied) needs to be done. This can be avoided with the Template Matching technique.

For the methods *average*, *matrix 1* and the ones using the *horizontal and vertical weight matrices*, the weighting between the sample based and block based intra prediction is chosen implicitly, i.e. it is same for all blocks or it is decided from the used prediction mode. That means that the choice of matrix does not need to be signalled from the encoder to the decoder and hence this extra signalling cost can be avoided. But for the method using the matrices 2-6 the matrix is chosen explicitly based on which gives least MSE which leads to that this choice needs to be signalled.

All the methods except the exhaustive have the same sample based intra prediction modes, which in an implemented version are supposed to be the ones chosen in HEVC. This means that the existing optimized signalling of the intra prediction modes is not changed. The prediction modes chosen in HEVC are optimized to not vary too much since this could generate a higher bitrate. But for the exhaustive method the intra prediction modes could vary a lot because the SBIP block and the BBIP are changed with respect to each other. This might generate a higher bitrate cost.

The computational complexity at the encoder side is quite high for all proposed methods, and most for the exhaustive method since it is an extensive algorithm. For the Template Matching method computations also needs to be done at the decoder side since the block is derived also here.

4.8.2 Comparing the methods

For all the methods, the combination of sample based and block based intra reduces the mean MSE in comparison with sample based intra, this holds for all tested sequences and block sizes except for the sequence 'Surfing' for N=8 (here the independent average has approximately 5% more mean MSE than sample based intra). Also, the combination seems to add structure and details which are missing in the sample based intra prediction. In most cases also less blocking artefacts are observed. Hence, these results suggests that a combined method of sample based and block based intra could improve the prediction.

The method that gives the most error reduction of all the proposed methods applied on the water test set is the *exhaustive* method when using horizontal and vertical weight matrix and testing all combinations of intra prediction mode with BBIP block. The method did also give very satisfying visual results. This indicates that a more extensive search of the combination of SBIP and BBIP should give better results, but one has to consider that the complexity is highly increased and that an extra signalling cost might arise due to more variation in the prediction modes (as mentioned when discussing the signalling cost in section 4.8.1). An alternative could be that the combination is tested for a number of the *most probable modes* and the best one of these is selected. This might also give less variation of the mode chosen than the exhaustive search could give (as mentioned in section 4.8.1) which would give better coding efficiency.

In table 4.12 and 4.13 the results for the methods average-dependent search, weight matrices 2-6 and the horizontal and vertical weight matrices with dependent search are shown next to each other. The average-dependent method seems to give almost the same reduction in error as when using matrices 2-6, which shows that the dependent method seems to reduce the error almost as much as when increasing the number of matrices. The method using the horizontal and vertical filter with dependent search gives even more error reduction, so an implicitly chosen weight matrix together with the dependent search seems to be a good combination. Comparing the independent and dependent visually shows that the dependent improves the edges more than the independent. The dependent method is searching for a block that fits best *together* with the sample based intra and hence it can reduce the error and visual artefacts more than the independent method which takes no consideration to the combined result. Also, the blocking artefacts on the right edge seems to decrease more with the dependent search according to the edgeMetric when testing the horizontal and vertical weight matrices.

Based on these results it would be to test all matrices 2-6 but choose these implicitly based on which prediction mode is used (as when using the horizontal and vertical matrices but divide the modes into more groups) and then use the dependent version of the search, i.e. choose the BBIP block based on which gives least error together with the sample based intra prediction block. This could give even better results since both an increased number of matrices used as well as the dependent search seems to give more error reduction. Due to lack of time this method was not tested in this work.

The refined method of Template Matching combined with the sample based intra prediction did unfortunately not show any improvements compared to the sample based intra prediction. This might be due to wrong values of parameters as discussed in this section (4.7), but it is a fact that the Template Matching probably does not provide as good match of the block as the IntraBC method since the template is used for the matching and not the actual block. But the Template matching is attractive due to it can reduce signalling for the block vector (although its more computationally demanding for the decoder), and therefore it would be interesting to investigate further.

The combined method seems to give better results than the sample based intra prediction especially when the sequence contains lots of texture. But when the sequence contains more plane areas it should probably be more efficient with only the sample based intra prediction since it is known to have good prediction for such areas.

So when a DC or planar mode is selected with the sample based intra prediction although the original sequence contains some complex texture, the combination of block based and sample based could improve the results. Also the sharp edges that emerge for these blocks could be reduced when having a weight matrix that improves the edges.

When comparing the results between blocksize N=32 and N=8 one has to consider that the used search areas are not of same size, since the search range was defined in number of blocks which was put to the same value but results in different areas. This could lead to that for N=8 the same structure is not found as when using N=32. For example the sequence 'Surfing' for which the dependent search gives 20% reduction in MSE when using block size N=32 but only $1-5\%$ reduction when using block size N=8. In general the results should be better for smaller blocksizes since the prediction is fitted more in detail, but when the search area is too small some structure might be missed out like for example a more coarse wave-structure. Also, the sample based intra prediction is predicting better for smaller blocksizes since all the samples in the block are relatively close to the reference samples, so to achieve an improvement might not be as important as for larger blocksizes. Although the smaller blocks 8x8 should give better results when the same search area is used, the signalling costs compared to 32x32 are 4 times larger.

Some observed trends are that for N=32 the gains compared to sample based intra are higher for higher QP, but for N=8 the gains seems to be lower for higher QP. There is no exact explanation to this but it might also be due to the fact that different search areas are used for the different blocksizes.

Table 4.12: Comparing the methods: average-dependent, matrices 2-6 and horizontal and vertical weight matrices - dependent. Blocksize N=32

N=32 %		**Average** **Dependent**			**Weigh matrices** **2-6**			**Hor & ver matrices** **Dependent**		
	QP	mean MSE	VIF	MSSIM	mean MSE	VIF	MSSIM	mean MSE	VIF	MSSIM
Ducks	22	-25.73	0.40	2.19	-24.29	1.13	2.03	-28.65	1.94	2.47
	27	-25.81	0.80	2.22	-24.93	1.45	2.10	-29.28	2.30	2.57
	32	-25.78	1.28	2.29	-24.38	1.77	2.10	-28.67	2.57	2.58
	37	-26.54	2.09	2.41	-24.30	2.30	2.18	-29.12	3.27	2.72
Drops	22	-19.05	0.19	0.36	-18.17	0.22	0.34	-20.03	0.28	0.41
	27	-19.03	0.20	0.37	-18.32	0.21	0.33	-19.79	0.33	0.41
	32	-20.18	0.34	0.39	-18.15	0.33	0.32	-20.58	0.45	0.43
	37	-20.15	0.53	0.43	-20.09	0.57	0.43	-20.89	0.58	0.47
Surfing	22	-18.46	-0.01	0.46	-16.62	0.41	0.41	-20.36	0.99	0.56
	27	-18.60	0.37	0.48	-17.77	0.76	0.46	-20.18	1.28	0.56
	32	-19.39	0.83	0.50	-18.27	1.18	0.49	-21.07	1.71	0.63
	37	-21.61	1.66	0.60	-20.58	1.81	0.57	-22.46	2.24	0.68
Riverbed	22	-19.92	0.18	0.43	-20.89	0.26	0.39	-21.13	0.40	0.48
	27	-20.03	0.24	0.42	-21.2	0.35	0.40	-21.38	0.49	0.46
	32	-21.3	0.43	0.45	-21.15	0.48	0.41	-22.66	0.65	0.50
	37	-22.71	0.63	0.5	-22.06	0.61	0.43	-23.23	0.77	0.53
CalmingWater	22	-25.07	0.35	0.51	-23.74	0.38	0.48	-25.86	0.53	0.57
	27	-25.07	0.43	0.50	-23.44	0.42	0.48	-25.87	0.57	0.56
	32	-25.31	0.56	0.50	-24.08	0.55	0.48	-26.18	0.69	0.56
	37	-26.41	0.75	0.55	-24.73	0.76	0.52	-26.60	0.89	0.57

Table 4.13: Comparing the methods: average-dependent, matrices 2-6 and horizontal and vertical weight matrices - dependent. Blocksize N=8

N=8 %		**Average Dependent**			**Weigh matrices 2-6**			**Hor & ver matrices Dependent**		
	QP	mean MSE	VIF	MSSIM	mean MSE	VIF	MSSIM	mean MSE	VIF	MSSIM
Ducks	22	-29.38	3.58	0.18	-29.29	3.48	0.17	-31.23	4.00	0.18
	27	-29.14	3.38	0.17	-29.62	3.38	0.17	-31.68	3.84	0.18
	32	-28.43	2.96	0.18	-28.82	2.84	0.17	-31.05	3.30	0.19
	37	-29.36	2.50	0.20	-28.75	2.34	0.18	-30.56	2.61	0.20
Drops	22	-18.48	0.67	0.04	-22.74	0.64	0.04	-23.29	0.73	0.04
	27	-18.85	0.61	0.04	-22.28	0.59	0.04	-22.80	0.66	0.04
	32	-19.46	0.56	0.04	-23.61	0.57	0.04	-23.30	0.58	0.04
	37	-17.50	0.42	0.04	-21.85	0.41	0.04	-21.85	0.45	0.04
Surfing	22	-4.00	0.77	0.03	-7.77	0.76	0.03	-4.66	0.85	0.03
	27	-4.52	0.72	0.03	-8.20	0.73	0.03	-5.31	0.80	0.04
	32	-2.26	0.54	0.03	-7.04	0.57	0.03	-3.01	0.55	0.03
	37	-3.75	0.59	0.02	-6.54	0.55	0.03	-1.05	0.52	0.02
Riverbed	22	-17.34	0.41	0.02	-22.42	0.42	0.03	-20.63	0.53	0.03
	27	-17.73	0.41	0.02	-23.14	0.44	0.03	-20.28	0.50	0.03
	32	-17.31	0.36	0.02	-23.37	0.40	0.03	-20.80	0.45	0.03
	37	-14.54	0.29	0.02	-18.64	0.29	0.03	-18.61	0.34	0.03
CalmingWater	22	-23.37	0.71	0.04	-26.51	0.72	0.05	-25.37	0.75	0.04
	27	-23.44	0.64	0.04	-26.06	0.64	0.05	-25.68	0.67	0.04
	32	-22.29	0.53	0.04	-24.37	0.50	0.04	-24.74	0.54	0.04
	37	-21.24	0.46	0.04	-23.62	0.44	0.04	-23.00	0.44	0.04

Chapter 5

Conclusions & future work

In this thesis some tests have been done in combining the sample based intra prediction (SBIP) and block based intra prediction (BBIP) in order to improve the sample based intra prediction (based on the current standard HEVC) for predicting difficult content, such as water. The methods have used weight matrices as well as the average when combining the methods and it has been shown that **a combination could decrease the MSE error as well as improve the VIF- and MSSIM-metrics**. The **visual results were also improved**; more details and less blocking artefacts were observed.

The method that seemed to give the best combination of block based intra and sample based intra when considering visual results, objective results as well as estimated signalling costs and computational complexity was the **dependent method when using horizontal and vertical weight matrices**. It would be interesting to investigate further if the use of more weight matrices, chosen implicitly depending on prediction mode, could generate an even better result.

To know if the proposed methods leads to an improvement of the total intra prediction process a more extensive investigation and implementation needs to be done, but with an improved prediction these methods seems promising.

When implementing one of the methods into the existing encoder, this method could be one of many being evaluated at different block sizes and the encoder then chooses the best method and block size by calculating the cost according to Rate-Distortion Optimization.

The methods proposed in this thesis were investigating some of the approaches that could improve the sample based intra prediction but there are several further investigations and improvements that could be done. Some proposed improvements are listed here below as possible future work:

- **Combine different blocksizes**
  A method that could be of interest would be to combine SBIP blocks and BBIP blocks of different sizes, for example using a 32x32 SBIP block and combining this with 4 BBIP of size 8x8. This could exploit the different advantages of the SBIP and BBIP and see which combination would be the best. This testing of different combinations of blocksizes should be done in the Rate-Distortion Optimization of the encoder when choosing the prediction mode.

- **Investigate Template Matching further**
  Investigate and optimize the parameters of the Template Matching method and also compare the results from this with the normal IntraBC to see whether it is worth the extra signalling in the IntraBC since it provides a better prediction.

- **Test more general weight matrices**
  The weighting of the SBIP and BBIP blocks were fixed matrices in this thesis. The weighting values were between 0 and 1, but this might not be optimal. A suggestion would be to use values between 1 and 0.5 for the SBIP and values between 0 and 0.5 for the BBIP, then the additional texture and detail that comes with the BBIP does not totally remove the (more conventional) SBIP. The sample based intra is also more based on the local structure than the block based intra might be, so it could be better to keep the sample based intra more than the block based intra.

  To make it even more general it would be interesting to optimize the whole weight matrix for each mode (similar as in [11] when combining inter and intra prediction).

- **Optimize against other metric than MSE**
  Since the Mean Squared Error is known for not having a good correlation with the Human Visual System (see section 2.4.1), it would be interesting if the matching of the BBIP could be done with any other better suited metric.

- **Add edges when matching the BBIP block**
  To see if the blocks chosen could fit better into the background and hence reduce the blocking artefacts it would be interesting to add some edges along the left and upper edge on the matching block (as the Template Matching template) in the IntraBC method.

- **Fractional search of BBIP blocks**
  Since it is a discrete grid of pixels there might sometimes be a more accurate match when looking for blocks also at sub-pixel level.

- **Do an extensive subjective evaluation**
  As a future investigation it would be interesting to do a more extensive subjective evaluation in order to confirm that the methods improve the visual quality.

- **Apply the method on several subsequent frames**
  In order to investigate the visual appearance when showing the sequence and not only a singe frame it would be interesting to apply the method on several frames in a sequence.

Appendix

Results from exhaustive search tested on standard sequences. Blocksize N=32, search range rl,rt,rr=3.

N=32		**Exhaustive**		
%	QP	mean MSE	VIF	MSSIM
BasketballDrill	22	-26.52	2.21	1.06
resolution 832x480	27	-26.58	2.42	0.99
	32	-26.88	3.08	1.11
	37	-24.93	3.55	0.93
BasketballDrillText	22	-29.39	2.35	0.85
resolution 832x480	27	-29.53	2.33	0.86
	32	-29.27	3.01	0.89
	37	-28.36	3.50	0.83
BasketballDrive	22	-24.01	3.57	0.82
resolution 1920x1080	27	-24.06	3.97	0.83
	32	-23.90	4.47	0.84
	37	-24.33	4.67	0.93
BasketballPass	22	-26.78	1.02	0.07
resolution 416x240	27	-27.09	1.08	0.14
	32	-27.54	1.20	0.19
	37	-26.13	1.69	0.39
BlowingBubbles	22	-28.62	0.70	0.06
resolution 416x240	27	-27.64	0.87	0.10
	32	-28.33	1.13	0.16
	37	-30.59	1.74	0.44
BQMall	22	-33.07	1.25	1.28
resolution 832x480	27	-33.19	1.51	1.28
	32	-32.87	1.60	1.35
	37	-33.91	2.35	1.47
BQSquare	22	-28.01	0.28	-0.16
resolution 416x240	27	-27.32	0.46	0.06
	32	-26.41	0.5	-0.14
	37	-25.42	0.57	0.01
BQTerrace	22	-21.98	2.76	1.56
resolution 1920x1080	27	-21.97	3.04	1.55
	32	-22.02	3.35	1.54
	37	-22.12	3.48	1.56

Cactus	22	-19.10	1.98	1.75
resolution 1920x1080	27	-19.29	2.31	1.65
	32	-19.40	2.64	1.66
	37	-19.96	3.26	1.75
ChinaSpeed	22	-14.96	0.89	0.73
resolution 1024x768	27	-14.94	0.81	0.66
	32	-14.58	0.93	0.69
	37	-15.29	1.39	0.92
FourPeople	22	-23.85	0.39	0.92
resolution 1280x720	27	-23.76	0.45	0.91
	32	-23.98	0.77	0.98
	37	-22.98	1.03	0.95
Johnny	22	-37.08	2.27	0.54
resolution 1280x720	27	-36.96	2.61	0.52
	32	-37.12	3.01	0.51
	37	-35.94	3.41	0.50
Kimono1	22	-39.78	5.28	2.36
resolution 1920x1080	27	-39.57	5.58	2.35
	32	-40.09	6.20	2.47
	37	-40.25	6.72	2.54
KristenAndSara	22	-24.88	0.82	0.36
resolution 1280x720	27	-24.86	0.85	0.35
	32	-25.09	1.40	0.37
	37	-24.66	1.30	0.41
ParkScene	22	-24.35	2.37	1.88
resolution 1920x1080	27	-24.62	2.76	1.94
	32	-25.18	3.32	1.95
	37	-25.20	4.14	1.93
PeopleOnStreet	22	-41.96	3.19	2.28
resolution 2560x1600	27	-41.95	3.61	2.27
	32	-41.81	4.13	2.23
	37	-41.88	4.78	2.24
SlideEditing	22	-26.49	2.66	2.10
resolution 1280x720	27	-26.42	2.84	2.19
	32	-26.33	3.02	2.14
	37	-26.75	3.28	2.20

Bibliography

[1] Gary J. Sullivan, Jens-Rainer Ohm, Woo-Jin Han, Thomas Wiegand. Overview of the High Efficiency Video Coding (HEVC) Standard. *IEEE transactions on circuits and systems for video technology*, 22(12):1649–1668, December 2012.

[2] Vivienne Sze, Madhukar Budagavi, Gary J. Sullivan. *High Efficiency Video Coding (HEVC), Algorithms and Architechture.* Springer International Publishing Switzerland, 2014.

[3] Johannes Ballé and Mathias Wien. Extended texture prediction for H.264/AVC intra coding. In *Proc. of IEEE International Conference on Image Processing ICIP '07*, volume VI, pages 93–96. IEEE, Piscataway, September 2007.

[4] S. L. Yu and C. Chrysafis. New intra prediction using intra-macroblock motion compensation. *JVT-C151, , Joint Video Team (JVT) of ISO/lEC MPEG and ITU-T VCEG Meeting*, May 2002.

[5] Satoshi Kondo, Hisao Sasai, and Shinya Kadono. Tree structured hybrid intra prediction. *IEEE Int. Conference on Image Processing ICIP 04*, 1:473–476, Oct. 2004.

[6] Thiow Keng Tan, Choong Seng Boon, Yoshinori Suzuki. Intra prediction by template matching. *IEEE Int. Conference on Image Processing ICIP 06, Atlanta, GA, USA*, Oct. 2006.

[7] Do-Kyoung Kwon and Madhukar Budagavi, Texas Instruments Inc. Fast Intra Block Copy (IntraBC) search for HEVC screen content coding. *IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 9–12, 2014.

[8] Do-Kyoung Kwon and Madhukar Budagavi, Texas Instruments Inc. Intra motion compensation and entropy coding improvements for HEVC screen content coding. *IEEE Picture Coding Symposium (PSC)*, pages 356–368, 2013, 8-11 Dec.

[9] Haoming Chen, Yu-Sheng Chen, Ming-Ting Sun, Ankur Saxena, Madhukar Budagavi. Improvements on Intra Block Copy in Natural Content Video Coding. *IEEE International Symposium on Circuits and Systems*, 2015.

[10] Kenneth Andersson. Combined Intra Inter Prediction Coding Mode. *ITU-T SG16/Q6 (VCEG), Doc. VCEGAD11, Hangzhou, China*, Oct. 2006.

[11] R. Cha, O. Au, X. Fan, X. Zhang, and J. Li. Improved combined inter-intra prediction using spatialvariant weighted coefficient. *IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–5, 2011.

[12] Zhenzhong Chen, Weisi Lin, King Ngi Ngan. Perceptual video coding: challenges and approaches. *International Congress on Mathematical Education (ICME), 2010*, pages 784–789, 2010.

[13] Yu-Bei Lin, Xing-Ming Zhang. Recent developments in Perceptual Video Coding. *Proceedings of the 2013 International Conference on Wavelet Analysis and Pattern Recognition*, pages 259–264, July 2013.

[14] Mathias Wien. *High Efficiency Video Coding (HEVC), Coding Tools and Specification.* Springer-Verlag Berlin Heidelberg, 2015.

[15] Merlin Paul, Abhilash Antony, Shreelekha G. Performance Improvement of HEVC Using Adaptive Quantization. *International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 1428–1433, 2014.

[16] Jani Lainema, Frank Bossen, Woo-Jin Han, Junghye Min, Kemal Ugur. Intra Coding of the HEVC Standard. *IEEE transactions on circuits and systems for video technology*, 22(12):1792–1801, December 2012.

[17] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, Eero P. Simoncelli. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, Apr. 2004.

[18] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh. Objective Video Quality Assessment. *Chapter 41 in The Handbook of Video Databases: Design and Applications*, pages 1041–1078, September 2003.

[19] David Salomon. *Data Compression, The Complete Reference.* Springer-Verlag London Limited, 4th edition, 2007.

[20] Marc Bosch, Fengqing Zhu, Edward J. Delph. Segmentation-Based Video Compression Using Texture And Motion Models. *IEEE Journal of selected topics in signal processing*, 5(7):1366–1377, Nov. 2011.

[21] Hamid R. Sheikh, Alan C. Bovik. A visual information fidelity approach to video quality. *The First International Workshop on Video Processing and Quality Metrics for Consumer Electronics*, pages 23–25, 2005.

Master's Theses in Mathematical Sciences 2016:E2
ISSN 1404-6342

LUTFMA-3286-2016

Mathematics
Centre for Mathematical Sciences
Lund University
Box 118, SE-221 00 Lund, Sweden

http://www.maths.lth.se/