ADDIS ABABA UNIVERSITY
ADDIS ABABA INSTITUTE OF TECHNOLOGY (AAiT)
SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

A VIDEO CODING SCHEME BASED ON BIT DEPTH ENHANCEMENT WITH CNN

By:
Daniel Getachew

Advisor:
Dr. Bisrat Derebssa

A thesis submitted in partial fulfillment of the requirements for the degree of Masters of Science in Computer Engineering

June, 2023
Addis Ababa, Ethiopia

ADDIS ABABA UNIVERSITY
ADDIS ABABA INSTITUTE OF TECHNOLOGY (AAiT)
SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

The undersigned have examined the thesis titled:

A Video Coding Scheme Based On Bit Depth Enhancement With CNN

Presented by Daniel Getachew, a candidate for the degree of Master of Science and hereby certify that it is worthy of acceptance.

Approved By Board of Examiners

Dr. Bisrat Derebssa	_____
Dean, SECE, AAiT, Thesis Advisor	Signature
Dr. Fitsum Assamenew	_____
Examiner I	Signature
Dr. Surafel Lemma	_____
Examiner II	Signature

Declaration of Authorship

I, Daniel Getachew, declare that this thesis titled, "A Video Coding Scheme Based On Bit Depth Enhancement With CNN" and the work presented in it are my own. I confirm that:

- This work was completed entirely or primarily while in candidature for a research degree at this university.
- This has been clearly stated if any part of this thesis has previously been submitted for a degree or other qualification at this University or any other institution.
- Where I have referenced the published work of others, this is always carefully acknowledged.
- I always give the source when I quote from other people's work. Except for such quotes, this thesis is entirely my work.
- I have acknowledged all main sources of help.
- Where the thesis is based on a cooperative effort by myself and others, I have specified clearly what was done by others and what I contributed.

Signed:

_____

Date:

_____

Abstract

Raw or uncompressed videos take a lot of resources in terms of storage and bandwidth. Video compression algorithms are used to reduce the size of a video and many of them have been proposed over the years. People also proposed video coding schemes which works on top of existing video compression algorithms by applying down sampling prior to encoding and restoring them to their original form after decoding for further bitrate reduction. Down sampling can be done in spatial resolution or bit depth.

This paper presents a new video coding scheme that is based on bit depth down sampling before encoding and use CNN to restore it at the decoder. However unlike previous approaches the proposed approach exploits the temporal correlation which exists between consecutive frames of a video sequence by dividing the frames into key frames and non-key frames and only apply bit depth down sampling to the non-key frames. These non-key frames will be reconstructed using a CNN that takes the key frames and non-key frames as input at the decoder.

Experimental results showed that the proposed bit depth enhancement CNN model improved the quality of the restored non-key frames by an average of 1.6dB PSNR than the previous approach before integrated to the video coding scheme. When integrated in the video coding scheme the proposed approach achieved better coding gain with an average of -18.7454% in Bjøntegaard Delta measurements.

Key words: bit depth down sampling, video coding, CNNs,

Acknowledgements

First I would like to thank my advisor, Dr. Bisrat Derebssa, for his consistent guidance and supervision. I would also like to express my heartfelt gratitude to my family for their valuable assistance in all aspects of my life. And above all I thank God for being by my side through all ups and downs.

Contents

List of Figures

List of Tables

Acronyms

Adam Adaptive moment estimation

AI Artificial Intelligence

AVC Advanced Video Coding

BD Bjontegaard Delta

BD-PSNR Bjontegaard Delta-PSNR

BD-Rate Bjontegaard Delta-Rate

CABAC Context-Adaptive Binary Arithmetic Coding

CAVLC Context-Adaptive Variable Length Coding

CNN Convolutional Neural Networks

CPU Central Processing Unit

CTC Common Test Condition

DCT Discrete Cosine Transform

DWT Discrete Wavelet Transform

EBDA-CNN Effective Bit Depth Adaptation Convolutional Neural Network

FCN Fully Connected Networks

GPU Graphics Processing Unit

HD High-definition

HEVC High Efficiency Video Coding

JVET Joint Video Experts Team

MSE Mean Square Error

PReLU Parametric Rectified Linear Unit

PSNR Peak Signal-to-Noise ratio

QP Quantization Parameter

RD Rate Distortion

ReLU Rectified Linear Unit

SSIM Structural Similarity Index Measure

SVD Singular Value Decomposition

Tanh Hyperbolic tangent function

Chapter 1

Introduction

A video is a series of frames displayed at a high rate, usually at 15, 30 or 60 frames per second, so we can perceive a smooth motion. A frame in a video is represented by array of pixel values which describes the brightness and color of the pixel. A raw video takes a lot of storage space and network bandwidth, for example just a minute long 8 bit color video with HD (720p) resolution and 30fps frame rate would need around 4.5 GB storage space and around 0.6 Gbit/s bandwidth to transmit. Even though storage capacity and communication technologies improved over the years we still need to shrink the size of the video to efficiently use them.

Video coding standards or CODECs are used to shrink the size of a video, and a lot of them have been developed since 1984 to achieve better compression efficiency than their predecessor. H.264/AVC [1], H.265/HEVC [2] are example of these standards and each of them achieves approximately 50% compression efficiency than the previous standard. These video coding standards exploit the redundancy existing in a video to better compress it. Two types of redundancies exist in a video, namely the spatial redundancy and the temporal redundancy. Spatial redundancy refers to the similarity existing between neighboring pixels with in the same frame and temporal redundancy refers to the similarity existing between consecutive frames.

Most video coding standards follow the hybrid video coding scheme [1, 2], which uses predictive coding and transform coding to compress a video sequence. Previously coded blocks are used to predict the current block in predictive coding. There are two types of predictions in video coding, intra prediction (prediction within the same frame) and inter prediction (prediction between different frames). The prediction errors or the residues are then transformed, quantized and entropy coded.

Most video coding standards use hand crafted algorithms designed carefully to give a better result for the components mentioned above. However due to the success of AI especially CNNs in image processing than hand crafted algorithm researchers are examining their use for video compression too.

Convolutional neural network (CNN) is a type of deep neural network algorithm which uses convolution operation rather than the classical matrix multiplication. They are usually used in image processing problems but they can be used in other domains too. They gain their popularity especially in image processing because of their ability to extract important features automatically, their reduced number of learnable parameters due to parameter sharing and their ability to capture spatial correlation exist in image data.

Recently CNNs found their way into the video coding domain. They have been used to improve the components used in video coding standards for example CNN based intra prediction, inter prediction, transform, and entropy coding etc. and some researchers also develop video compression algorithms entirely using CNNs.

Down sampling (reducing spatial resolution) prior to encoding and up sampling it to the original resolution after decoding can be used to further reduce the bitrate in video coding. This technique firstly used for low bitrate compression but due to the emergence of powerful supper resolution algorithms, especially using CNN, now this technique can be used for high bitrate compression too.

Similar to down sampling in spatial resolution another technique used in video coding is effective bit depth (the actual bit depth used to represent the image) down sampling (reduction) prior to encoding and restore the full bit depth after decoding. Simple quantization or tone mapping can be used to reduce the effective bit depth of the image. The method we are proposing belongs to this group, we aim to develop a video coding scheme based on effective bit depth down sampling but different from previous methods we will consider and exploit the temporal correlation which exists between consecutive frames of a video sequence.

1.1 Problem statement

Many bit depth enhancement algorithms have been proposed over the years mainly to display existing low bit depth images on monitors with high bit depth. These algorithms try to suppress false contour and color distortion introduced by simple de-quantization of low bit depth to high bit depth. Some researchers also developed bit depth enhancement algorithms to use them in video coding schemes based on bit depth down sampling (bit depth reduction).

Most of the bit depth enhancement algorithms proposed are for single image bit depth enhancement and they forget the temporal correlation exist in a video sequence. Even the bit depth enhancement algorithms used in video coding schemes treat each frame individually. To the best of our knowledge there is one algorithm which considers the temporal correlation. However this algorithm only tries to produce consistent consecutive frames.

The purpose of this study is to develop a video coding scheme based on bit depth reduction which exploits the temporal correlation that exists between consecutive frames of a video sequence. In this scheme some key frames which are compressed without bit depth reduction will be used to enhance the bit depth of the frames that are compressed with their bit reduced.

1.2 Objectives

1.2.1 General objective

The main objective of this study is to develop a video coding scheme which has a better bitrate reduction than existing video coding algorithms. We will develop a video coding scheme based on bit depth reduction and we will exploit the temporal correlation exists between consecutive frames.

1.2.2 Specific objectives

- To develop a bit depth enhancement algorithm which exploits the temporal correlation exist between consecutive frames in a video.
- To integrate this algorithm to existing video coding standard for better bitrate reduction.
- To measure how much bitrate reduction we can achieve using this technique.

1.3 Research questions

This study aims to find answers for the following research questions (RQ).

RQ1. Does the temporal correlation that exist between consecutive frames in a video improve the quality of the reconstructed frame in CNN based bit depth enhancement algorithms?

RQ2. Can we achieve more bitrate reduction by using this proposed bit depth enhancement CNN model in video coding schemes which are based on bit depth down sampling?

1.4 Scope

We want to exploit the temporal correlation exists between consecutive frames of a video sequence for video coding schemes which are based on bit depth down sampling in our study, however this might not be true for some video sequences because of a high global motion or scene change in the video and this may affect the performance of our method. Therefore in this study we only consider videos with highly correlated consecutive frames for example security camera footages and video conferencing, in general videos with low global motion or scene change.

1.5 Significance of the study

This study aims to develop a video coding scheme that has a better bitrate reduction than existing video coding schemes that are based on bit depth down sampling. Furthermore the bit depth enhancement CNN model developed can be used in assisted bit depth enhancement applications to increase the quality of low bit depth media contents.

1.6 Contribution

This study aims to assess the impact of considering the temporal correlation which exists in consecutive frames of a video sequence in video coding schemes that are based on bit depth down sampling. In doing that these are the main contributions of this research work.

- Developed a video coding scheme based on bit depth down sampling which takes into consideration the temporal correlation which exists in consecutive frames of a video sequence.
- Developed a CNN model that takes key frames with no bit depth reduction and non-key frames with bit depth reduction as input to restore the bit depth of the non-key frames.
- Assessed the impact of considering the temporal correlation which exists in consecutive frames of a video sequence in video coding schemes that are based on bit depth down sampling

1.7 Paper organization

The thesis document is organized in six chapters: The first chapter presents the motivation, the general and specific objectives, the scope, and the contribution of the thesis. The second chapter presents the details of theoretical backgrounds of concepts related to this study. The third chapter presents in detail the exploration of previously done related works of literature. The chapter four describes the proposed methodology. The fifth chapter presents the experiment setups of the entire thesis and gives a detailed explanation of the results obtained from the conducted sets of experiments. The final chapter concludes the thesis and indicates future research directions.

Chapter 2

Theoretical background

2.1 Video codding

Raw or uncompressed video takes a lot of resources in terms of storage and bandwidth. To utilize these resources efficiently the size of the video needs to be reduced. Video compression is a method that's used to reduce the size of a video. Even though network and storage capacities increase drastically in recent years video compression is still needed to use them efficiently.

Data compression usually works by removing redundancy from it. Data can be compressed without any loss by carefully removing redundancies, however currently existing lossless compressions achieve small amount of compression when used for video compression. Due to this most video compression techniques follows on lossy compression, which achieves grater compression with a cost of losing some information.

Videos have redundancy in spatial, temporal and frequency domains and video compression algorithms exploit this properties to better compress the video. Spatial redundancy refers to the similarity existing between neighboring pixels with in the same frame and temporal redundancy refers to the similarity existing between consecutive frames. Redundancies in frequency domain can be removed due to the fact that human eye and brain are more sensitive to lower frequencies, and the high frequencies can be removed without affecting the quality drastically [1].

Most video coding standards follow the hybrid video coding scheme [1, 2], which uses predictive coding and transform coding to compress a video sequence. Previously coded blocks are used to predict the current block in predictive coding. There are two types of predictions in

video coding intra prediction (prediction within the same frame) and inter prediction (prediction between different frames). The prediction errors or the residues are then transformed, quantized and entropy codded.

2.1.1 Intra prediction

Intra prediction tries to reduce the spatial redundancy in a video. A block in a frame is predicted using another previously coded block within the same frame. In most cases the upper and left blocks from the current block as shown in Figure 2.1 are used for intra prediction since they will be already decoded and ready to predict the current block. Video coding standards usually have more than one intra prediction modes and the best one for a given block will be selected based on the rate distortion cost.

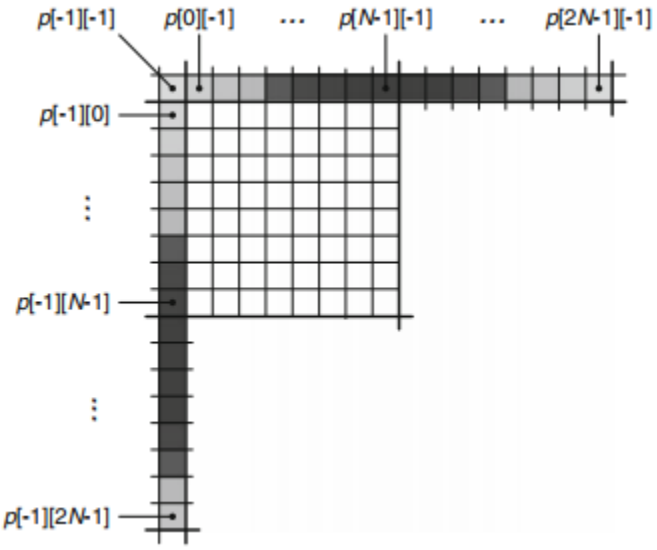

Figure 2.1: Example of reference samples for intra prediction

2.1.2 Inter prediction

Since videos are images or frames displaying at high rate a lot of similarities exist between consecutive frames. This property is called temporal redundancy and the inter prediction step tries to remove this redundancy. Inter prediction achieve this by using a process called motion estimation and compensation.

The frames in a video are divided into group of pictures and each group contains 3 types of frames namely I-frames, P-frames and B-frames. I-frames are compressed without inter prediction they are used as reference to other frames in the group. P-frames are predicted

using previous I or P frame as a reference. B-frames on the other hand are predicted using one previous and one future frame as a reference.

The inter prediction step starts by dividing the frame into smaller blocks called macroblocks. Then the block is searched in the reference frame(s) by comparing the current block with each block in the reference frame(s). Then a motion vector is generated which represents the movement of the block from one position to another as shown in Figure2.2. This process is called motion estimation.

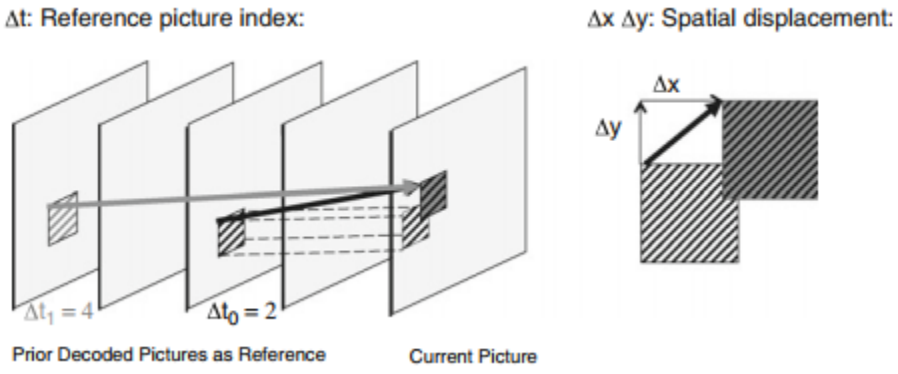

Figure 2.2: Inter frame prediction

After the motion vectors are generated the current frame is predicted by applying the motion vector to the reference frame(s). This process is called motion compensation. After the current frame is predicted only the motion vectors and the prediction error or residual is transferred to the next step. The residual is transferred to the next step because the residual has very low information than the frame itself which leads to better compression.

2.1.3 Transform coding

After inter prediction is done the residual frame is transformed into frequency domain in this phase. The frames are transformed into frequency domain to take advantage of how human eye and brain work. Our eye is sensitive to the low frequency components and the high frequency components can be removed without affecting the quality of the frame.

The transform coding can be applied to the entire frame or block by block by dividing the frame into smaller blocks. Most video compression standards use the block based transform. A transformation like DCT, DWT, and SVD function is applied to each blocks of the frame in this stage. The transform step is reversible which means it is lossless, it just help us to identify which part to throw away.

The higher frequency components will be removed in the quantization step where the coefficients calculated in the transform step are divided with an integer to map the range of the transformed values to reduced range of values. The quantized frame can be represented with fewer bits since the range is reduced.

After the transformed coefficients are quantized the resulting vector will have some non-zero coefficients and a lot of zero coefficients [1]. These values will be reordered and the zeros will be encoded efficiently before going to the next step for better compression.

2.1.4 Entropy coding

In this stage the data that represents the video is converted into its compact form called bit stream. The data includes quantized coefficients, motion vectors, prediction modes etc. Entropy coding techniques exploit the statistical redundancy exist in the data usually by representing coefficients with high occurrence with less number of bits [1].

2.2 Convolutional neural networks

Convolutional neural network (CNN) is a type of deep neural network algorithm which uses convolution operation rather than the classical matrix multiplication. They are usually used in image processing problems because they work with grid-structured inputs [3] but they can be used in other domains that has a grid-structured inputs too.

Convolutional neural networks differ from other neural network algorithms mainly because of the use of convolution operation. A convolution operation is a dot product between a weight matrix and a corresponding matrix taken from the input. This is important to capture spatial correlation exist in the input data.

The architecture of convolutional neural networks is inspired by how the visual cortex in the brain works [3]. The cells in the visual cortex are sensitive to small part of the visual field called receptive field that is why convolutional neural networks have sparse connection from one layer to the next. These cells are also excited based on the distinct patters of the object in the visual field that's why different filters are used in convolutional neural networks.

2.2.1 Why CNN?

One of the main reasons Convolutional neural network perform better than traditional methods, especially in image processing applications, is its ability to extract important features automatically without any human help.

Convolutional neural network is able to learn because the convolution operation is applied by sliding a small sized filter over the input feature map, due to that convolutional neural networks have reduced number of learnable parameters, this is called parameter sharing. This is an important property for two reasons, first since the parameters are shared throughout the feature map, a filter that detects a distinct feature can detect its existence irrespective of the position in the feature map. Parameter sharing also leads to small number of parameters which reduces the complexity of the network.

Convolutional neural network work with grid-structured inputs, so they maintain the spatial relationship exist between the neighboring values in the input data. In fully connected neural networks the input data needs to be flattened before going in to the network which destroys the spatial correlation exists in the input data, this will not be a problem for convolutional neural network because it works with grid-structured inputs.

2.2.2 Basic structure of a convolutional network

Convolutional neural network works more like other feed forward network, however the operations in each layer are spatially organized and the connection between layers is sparse. Convolutional neural network mainly consists of convolution layer, pooling layer and activation layer. It may also have fully connected layers at the final stages to produce the output. Figure 2.3 shows the basic structure of a convolutional neural network.

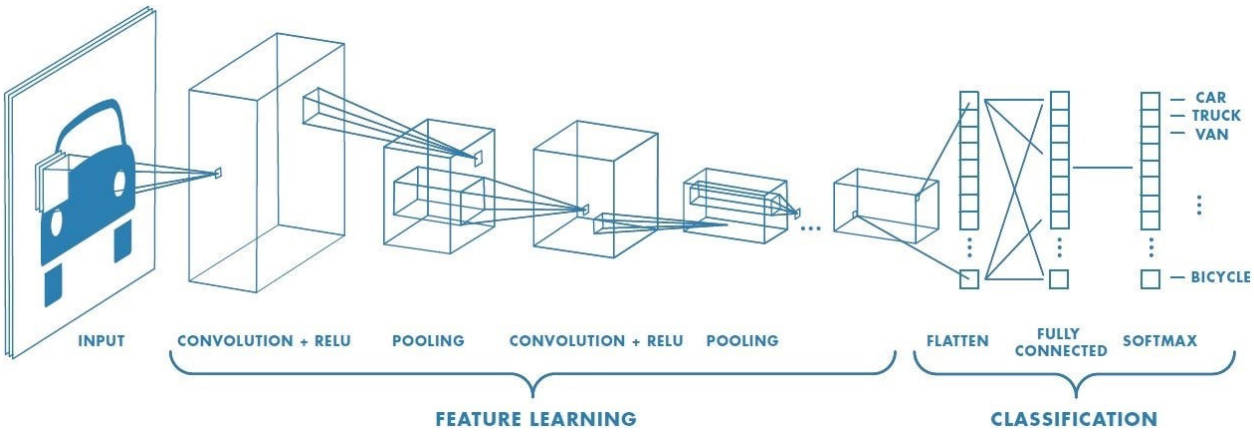

Figure 2.3: Basic structure of a convolutional neural network

Convolution layer

In convolutional neural network the parameters are organized in 3 dimensional structures called filters or kernels. In the convolution layer a convolution operation is performed by

placing the filters in each possible positions of the input and performing the dot product between the two to produce what we call feature maps in the next layer. Many different filters can be used in a convolution layer to produce different feature maps that recognize different patterns in some part of the input.

To define the convolution operation formally let us assume the pth filter in the qth layer has parameters denoted by the 3-dimensional tensor $W^{(p,q)} = [w_{ijk}^{(p,q)}]$ with size Fq X Fq X dq and i, j, k represents the position of a parameter in the filter. The feature maps in the qth layer are also represented by the 3-dimensional tensor $H(q) = [h_{ijk}^{(q)}]$ with size Lq X Bq X dq. Then the convolution operation is defined by the following Equation 2.1.

$$h_{ijp}^{q+1} = \sum_{r=1}^{F_q} \sum_{s=1}^{F_q} \sum_{k=1}^{d_q} w_{rsk}^{(p,q)} h_{i+r-1,j+r-1,k} \qquad \begin{array}{l} \forall i \in 1\ldots., Lq - Fq + 1 \\ \forall j \in 1\ldots., Bq - Fq + 1 \\ \forall p \in 1\ldots., dq + 1 \end{array} \tag{2.1}$$

The convolution operation reduces the size of the feature maps going from the q^{th} layer to the $(q+1)^{th}$ layer because there are smaller number of possible positons the filter can be aligned to the input without the portion of the filter sticking out of the input. This property is not desirable because it leads to loss of information at the borders of the input. Padding can be used to avoid this problem. Padding adds a number of pixels all around borders of the feature map to maintain the spatial resolution in the next layer. The values of this padded features is usually set to 0 so that they have no contribution in the result of the final dot product.

Activation layer

Activation layer is applied to introduce non linearity in the network. It takes feature maps from the convolution layer and transform them into activation maps using some activation function. Different activation functions like sigmoid or Tanh can be used in this layer, but ReLU is most commonly used in convolutional neural network because it has a better performance than the other activation functions.

Pooling layer

The pooling layer is used to increase the receptive field of a layer while reducing the spatial resolution. Receptive field refers the part of the input that activates this feature. Larger receptive field are desirable because they capture larger portion of the input.

A pooling operation is performed in small region of size P_qXP_q on each feature maps in the layer independently. For each P_qXP_q region in the feature map the maximum in that regain is taken as the output of the pooling operation. This method is called max-pooling, however this is not the only type of pooling operation, for example one can use average-pooling, which takes the average in the P_qXP_q region to produce the output, for the pooling operation.

2.3 Bit depth enhancement

Bit depth refers to the number of bits required to represent a single channel of a pixel. The bit depth used for most media sources is 8 bit, however there exists monitors that supports high dynamic ranges with a bit depth of 10, 12 or 16 bits. Bit depth enhancement is a technique used to increase the bit depth of a media content to increase its quality. Different techniques including zero padding, inverse tone mapping, bit replication, hand crafted mathematical models and currently deep learning-based methods can be used for bit depth enhancement.

Bit depth enhancement techniques can also be used in video coding schemes that are based on bit depth down sampling. In these video coding schemes the actual bit depth used to represent each channel of a pixel is reduced before encoding and will be restored at the decoder by using bit depth enhancement techniques.

Chapter 3

Literature review

As it is mentioned above most widely used video coding standards use handcrafted algorithms that are carefully designed to give better result. However researchers have been using CNNs recently to replace some tools used in video coding standards or to develop video compression algorithms fully using CNNs. Most related to our method researchers also used CNNs for image and video bit depth enhancement and video compression algorithms based on bit depth reduction. This section presents a review of these approaches.

3.1 Hand crafted video compression standards

H.264 [1] and its successor HEVC [2] are state of the art video coding standards currently. Even though HEVC gives approximately 50% bitrate reduction than H.264 with same quality, because of its complexity H.264 is still used in many applications. These standards follow the hybrid video coding scheme and use handcrafted algorithms for the different components mentioned above.

3.1.1 Intra prediction

H.264 has 9 modes for intra prediction of 4x4 luma blocks 4 modes for 16x16 luma blocks and all chroma blocks. These modes predict a block by extrapolating, interpolating or averaging samples of the top and left blocks of the current block. HEVC introduce 35 intra prediction modes for all block sizes for better prediction. HEVC also uses reference sample substitution to allow using every modes when there are an available reference samples which is not possible in H.264 in which only certain modes can be used if there are unavailable reference samples.

3.1.2 Inter prediction

Both standards don't specify a particular motion estimation algorithm to be used and the implementer can choose any algorithm. Both standards also support fractional motion vectors to quarter pixel precision for luma components and 1/8 pixel precision for chroma components. This fractional samples can be calculated by interpolating neighboring integer or fractional pixel samples. In H.264 half pixel samples are calculated by interpolating neighboring integer samples using a 6tap filter and quarter pixel precision are calculated by first calculating half pixels samples and then averaging them. HEVC uses 7/8 tap filters for all pixel precisions and don't have to calculate half pixel precisions first to calculate quarter pixel precisions.

3.1.3 Transform and quantization

H.264 uses the approximation of 4x4 Discreet cosine transform to avoid mismatch due to different approximations and for ease of implementation. In order to have the properties of DCT, which are lost due to the approximation, H.264 uses different quantization matrices for different quantization steps. After a 4x4 core transform the DC coefficients of intra predicted 16x16 luma blocks and all chroma blocks will be transformed using Hadamard transform and transmitted with the core transform. HEVC also use the approximation of DCT carefully designed to have the important properties of DCT hence there is no need to use different quantization table for different quantization steps in HEVC. HEVC allows transform using different size from 4x4 to 32x32 and also specifies an alternate 4x4 transform based on discrete sine transform.

3.1.4 Entropy coding

For entropy coding H.264 uses Context-Adaptive Variable Length Coding (CAVLC), where variable length codes are chosen based on context and Context-Adaptive Binary Arithmetic Coding (CABAC), which was first introduced in H.264, it is based on arithmetic coding but the probability of each symbol is estimated based on some context. HEVC uses improved version of CABAC.

3.2 Video coding using CNNs

As it is mentioned above CNNs have been used in video coding to improve the performance of some components of traditional coding standards, CNNs have been used in intra prediction,

inter prediction, transform and quantization or in entropy coding. There are some algorithms also which used CNNs entirely for video coding.

Cui et al. [4] and Wang et al. [5] propose a CNN to refine the prediction of HEVC to minimize the prediction error. Cui et al. [4] follow a residual learning technique and their CNN takes the best prediction of the current block and its three nearest blocks as input to produce the prediction. Wang et al. [5] propose a CNN that extract feature at different scale of the input to take advantage of multi scale feature maps and they also use multiple lines of reference pixels from neighboring blocks. Dumas et al. [6] propose a neural network based intra prediction, they used a fully connected networks (FCN) to predict small blocks and a CNN to predict large blocks. Video coding standards selects the best partitioning mode and intra prediction mode based on the rate distortion optimization, which tries all possible modes and selects a mode with minimum rate distortion cost. Trying all possible combination of modes is complex with the complexity increasing with number of modes available to choose. To address this problem Laude et al. [7] and Song et al. [8] propose a CNN for intra prediction mode decision and Liu et al. [9] propose to use CNN for portioning mode decision.

CNNs have been also used for inter prediction recently. Yan et al. [10] propose a CNN to generate half pixel samples from integer pixels to replace the fixed 8 tap interpolation filter used in HEVC. Zhang et al. [11] also introduce a CNN to generate half pixel samples but in their network the prediction and the residual are used as inputs instead of the reference block. Yan et al. [12] again propose multiple CNN models to generate different combination of half and quarter pixel samples. Zhao et al. [13] used a CNN to combine the forward and backward predictions in a bi directional motion compensation instead of the linear combination of the two predictions used in HEVC. To refine the prediction produced by HEVC inter prediction Huo et al. [14] propose a CNN which considers spatial correlation exist between adjacent blocks and takes some reference pixels from adjacent reconstructed blocks along with the prediction as an input. Lee et al. [15] propose a block matching algorithm for motion estimation, their algorithm uses representative matching to match blocks but instead of taking representative values directly from the blocks they take them from features extracted by a CNN. Choi et al. [16] devise a different approach for inter prediction which doesn't need motion information to be transmitted by using a CNN that can predict the current frame using two reference frames. They integrated their algorithm in HEVC additional to HEVC's inter prediction algorithm and the encoder will select the best one based on RD cost for inter prediction. Some researchers, Lee et al. [17], Zhao et al. [18] also used CNNs to generate virtual reference frames which can be used as reference

frames along with conventional reference frames for inter prediction.

Some researchers also consider using CNNs in transform, quantization and entropy coding. Liu et al. [19] propose a CNN based transform where CNNs are used for transform, quantization and inverse transform. They trained their CNNs jointly to minimize the RD cost and used the transform network at the encoder and the inverse transform network at the decoder. Alam et al. [20] develop a quantization strategy where a quantization parameter (QP) is chosen based on artifact visibility threshold predicted by a CNN. Ma et al. [21] propose a CNN for probability estimation of a given syntax element to assist the arithmetic coding for entropy coding, where in HEVC the CABAC uses different manually designed context models for probability estimation.

In addition to using CNNs to enhance the efficiency of different components in the traditional hybrid video coding they have been used to develop end to end deep video compression algorithms. Chen et al. [22] propose to use CNN to compress the residual signal, every block is first predicted using inter or intra prediction. For inter prediction they use traditional motion estimation and compensation, and for intra prediction the used a CNN to compress the block. The residue from both inter and intra prediction is also compressed using another CNN. Wu et al. [23] propose a video compression algorithm based on image interpolation. They first compress key frames using deep image coding scheme and use these frames to interpolate intermediate frames. The interpolation is performed using a CNN and to assist the interpolation, motion information and the prediction error are used as an input to the interpolation network. Lu et al. [24] used CNNs to develop a video coding scheme based on inter prediction. They replaced every components in the traditional video coding scheme. In their scheme they used a CNN to estimate the motion between two frames and another CNN for motion compensation, to predict the current frame using previous frame and motion data. The motion information and the prediction error will be compressed again using a different CNN. All the networks are jointly trained to minimize the rate distortion cost. Chen et al. [25] propose another method for end to end deep video coding. In this method they use two previous frames and previously coded neighboring blocks to predict the current block using CNN. The residue is then compress using a CNN also.

3.3 Bit depth enhancement

We are proposing a video coding scheme based on effective bit depth reduction prior to encoding and recover it to the original bit depth after decoding. Bit depth enhancement algorithms have been proposed specially to make low bit depth contents display on high

bit depth monitors for a better quality. This is an image processing problem and as in other image processing problems CNN based approaches perform better than hand crafted approaches in this area too.

Liu et al. [26] are the first to use deep learning approach for bit depth enhancement. They propose a network with transposed convolutional layers (de-convolutional layers) because of their ability to capture high-order image structure beyond edge primitives. Liu et al. [27] also propose an auto encoder like network which has convolutional layers followed by de-convolutional layers for bit depth enhancement. Every two layers are connected in their network to solve the gradient vanishing problem and they follow residual learning to make the training easy. Peng et al. [28] also propose an auto encoder like network with convolutional layers followed by de-convolutional layers, but they introduce a new loss function, range loss, which tries to keep pixel values within range to address color distortion problem. Zaho et al. [29] claims existing approaches tries to remove artifacts and not to recover the list significant bits accurately. To solve this problem they propose a two channel network for flat areas and non-flat areas of the image and combined them at the end to produce the high bit depth image. The flat area is first smoothed before entering the network to give better result.

The bit depth enhancement algorithms mentioned above are designed for image bit depth enhancement and they forget the temporal correlation exists in a video sequence and usually produce inconsistent consecutive frames. Liu et al. [30] propose a bit depth enhancement network for videos. Their network takes five consecutive low bit depth frames as input, two motion compensated previous frames and next frames in addition to the current frame to produce consistent consecutive frames.

Nguyen et al. [31] propose bit depth reduction prior to encoding to compress depth map in 3D video compression. They use weighted mode filtering, hand crafted algorithm, to reconstruct the depth map with the original bit depth after decoding. Zhang et al. [32] also propose a video coding scheme based on bit depth reduction but they use a CNN with skip connection and residual blocks to reconstruct each frames individually to their original bit depth. They trained different models for different QP values and each frame is divided in 96x96 blocks as input to the CNN and combined at the output after all of them are reconstructed. Ma et al. [33] also used a CNN to restore the original bit depth but they adopt a generative adversarial network (GNA) to train their CNN.

CNNs show a promising result in the video coding domain. They have been used to develop the tools used in video coding standards or to develop video coding algorithms that are entirely based on them and show their superiority over hand crafted algorithms. They have

also shown grate results in the bit depth enhancement problem and video coding schemes which are based on these algorithms. However these algorithms treat each image individually and often forget the temporal correlation exists in a video sequence. We believe the results can be improved if we exploit this correlation.

Chapter 4

Proposed approach

The proposed video coding scheme is shown in Figure 4.1 below. To exploit the temporal correlation that exists in consecutive frames of a video we introduce the concept of key frames and non-key frames. Key frames are those frames that will be encoded without their bit reduced and will be used to extract information that can help us in restoring the bit depth of the non-key frames which are encoded after their bit depth is reduced. The coding process starts by splitting the frames in to key and non-key frames, the key frames are encoded without their bit depth reduced and non-key frames are encoded after their bit depth reduced (3 bits in this paper) by bit shifting. After the bit depth of the non-key frames is reduced both the key frames and non-key frames are then encoded using an existing encoder. At the decoder the key frames and non-key frames are reconstructed first using the existing decoder and the bit depth of the non-key frames is restored using the bit depth enhancement CNN.

4.1 The bit depth enhancement CNN

The architecture of the bit depth enhancement CNN is shown in Figure 4.2. The architecture is very similar to the EBDA-CNN that is used in the effective bit depth adaptation in video coding. However it is different in such a way that it takes the key frames as input in addition to the non-key frames, which are to be restored. The CNN is designed to take 256X256 key frame and non-key frame (with reduced bit depth) and produce non-key frame with restored bit depth as an output. The proposed model follows a structure like ResNet by using residual blocks to solve the vanishing gradient problem which arises when the network becomes deep. First the key frames and non-key frames are fed to independent convolutional

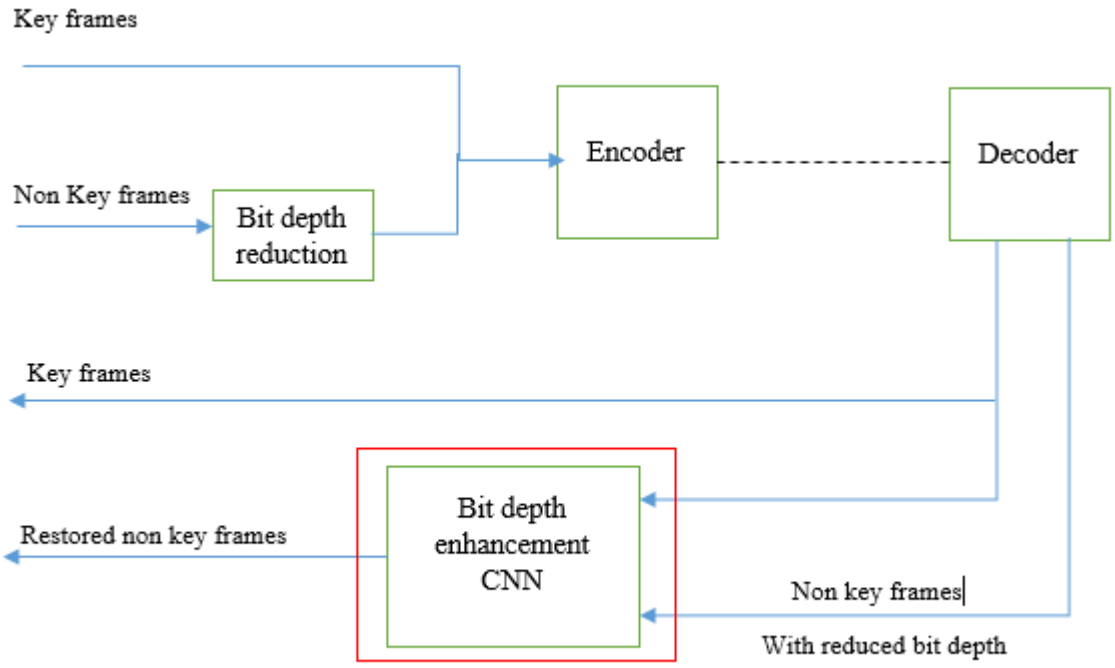

Figure 4.1: The proposed video coding scheme

layers to produce independent feature maps. Then the outputs are concatenated and used as input for the next convolutional layer. This layer is followed by 16 residual blocks which consist of 2 convolutional layers with skip connection from the input to the output and used PReLU as activation function. Finally after one convolutional layer and a Tanh activation the input non-key frame is added to the output of this layer to produce the final output. All the convolutional layers have 3x3 kernels and 64 feature maps with a stride of 1 with same padding to restore the resolution. There is no pooling layer in the model. And the loss function used is L1 loss. The existing bit depth enhancement model for video compression [32], which takes only non-key frames as input was also built for comparison.

4.2 Dataset collection and preparation

Raw videos and some compressed videos were collected from publicly available databases (YUV21 [34], harmonic 4k footages [35], UVG [36], xiph.org [37]). These videos have different resolution but they are reduced to 256x256 to reduce memory cost.

The inputs and the target for training the bit depth enhancement CNN are the compressed

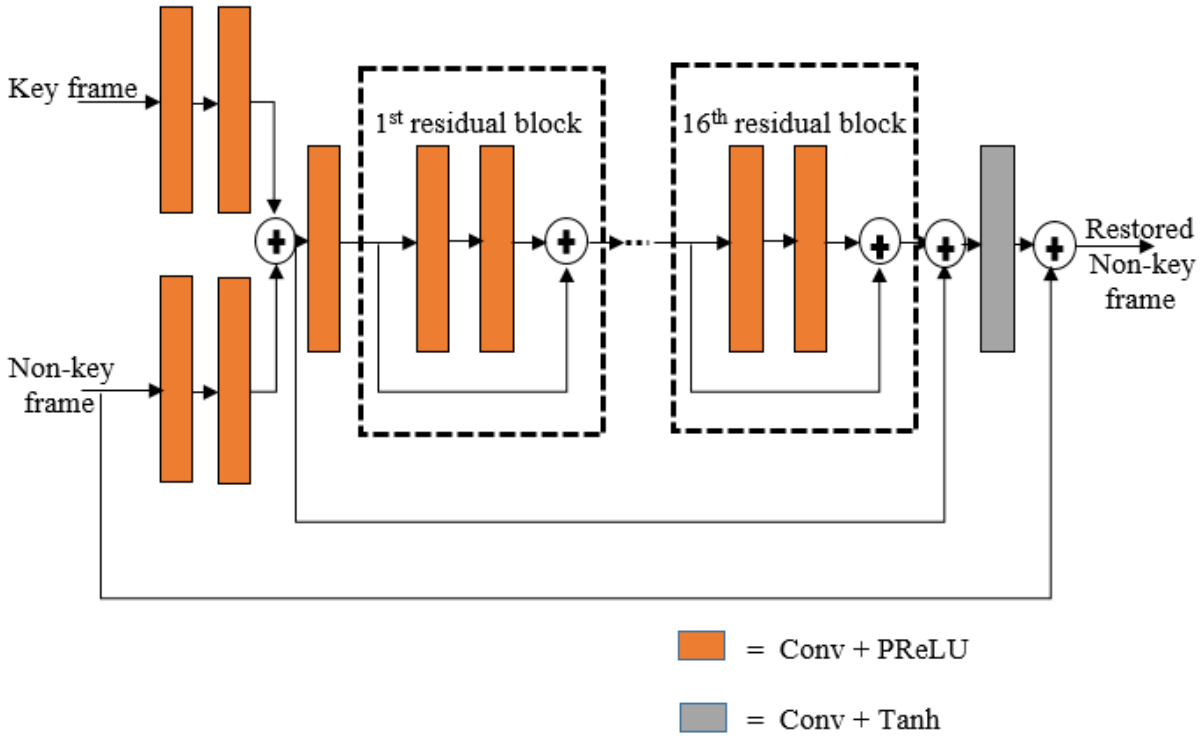

Figure 4.2: The proposed bit depth enhancement CNN model

key and non-key frames and original frame respectively. The videos were compressed using x265 (HEVC encoder) by reducing the bit depth of every non-key frames. The decoded and original videos then converted from yuv420 format to yuv444 format. The training examples were generated by selecting the first frame as key frame and randomly selecting 3 frames in every 10 consecutive frames in the compressed video as non-key frames and the corresponding 3 frames from the original video.

Both CNN models for bit depth enhancement were built and trained on 5220 examples (including 4220 for training and 1000 for testing). The hyper parameters employed in the training are Adam optimization [38], batch size of 4, 25 epoch and a decaying learning rate with initial value of 0.001.

Chapter 5

Experiment

In order to answer our research questions, we conducted a series of experiments. This section describes the experimental setup followed and evaluation methods used to evaluate the performance of the proposed approach.

5.1 Experimental setup

To evaluate the performance of the proposed approach and answer our research questions we conducted two sets of experiments.

Experiment 1: this experiment aims to evaluate the performance of the proposed bit depth enhancement CNN in restoring the bit depth of the non-key frames. In order to do that we built the proposed model and the EBDA-CNN model and compared them using the test dataset by measuring the quality of the restored non-key frames. This experiment was designed to answer the first research question. In this experiment we want to find out which CNN model performs better in restoring the frames that have their bit depth reduced before we integrate them into the video coding scheme.

Experiment 2: this experiment aims to evaluate the performance of the proposed bit depth enhancement CNN when integrated to the proposed video coding scheme. The first set of experiments try to evaluate the performance of the proposed approach with respect to N where N is the number of frames between two key frames. This experiment compares the proposed approach and EBDA-CNN on the quality of the decoded and restored video sequences. Then the second experiment aims to evaluate the performance of the proposed approach in terms of bitrate reduction for the best performing N value.

5.1.1 Test data

In this study two sets of data were used to evaluate the performance of the proposed approach in restoring the bit depth of the non-key frames and bitrate reduction. The dataset collected from the publicly available databases were used to test the performance of the proposed approach in restoring the bit depth of the non-key frames. We divide the dataset into training and testing with a ratio of 80% to 20% respectively.

To evaluate the performance of the proposed approach in terms of bitrate reduction, test sequences from JVET CTC were used. The data from JVET CTC used in this test are shown in the following Table 5.1.

Video sequence	Length (frames)
BasketballDrill	500
BQMall	600
PartyScene	500
BasketballDrive	500
Cactus	500
Kimono1	240
ParkScene	240
BasketballPass	500
BlowingBubbles	500
BQSquare	600
RaceHorses	300
FourPeople	600
Johnny	600
KristenAndSara	600
PeopleOnStreet	150
Traffic	150

Table 5.1: Test video sequences

5.1.2 Experiment environment

In this study google colaboratory was used to train and test the bit depth enhancement CNNs. We used the free version of google colaboratory with GPU run time with specifications shown in the Table 5.2 below, this may vary due to the availability of resources.

GPU	Tesla T4
GPU Memory	12GB
Processor	Intel Xeon CPU @2.20 GHz
Available RAM	13GB
Operating system	Ubuntu 20.04.5 LTS

Table 5.2: Hardware and software specification used for training and testing the bit depth enhancement CNNs

5.2 Evaluation metrics

In this study the PSNR and SSIM evaluation metrics are used to evaluate the quality of the restored frames and the Bjøntegaard Delta (BD) measurement is used to evaluate the bitrate reduction between the two approaches.

5.2.1 PSNR

Peak Signal-to-Noise Ratio PSNR [39] is widely used objective image quality evaluation metrics. PSNR refers to the ratio between the maximum possible value of a single pixel and the noise that affected it. The mathematical formulation of PSNR is defined in Expression 5.1.

$$PSNR(f,g) = 10 * log_{10}(MAX/MSE) \tag{5.1}$$

Where MAX is the maximum possible value of a single pixel and MSE is defined in Expression 5.2.

$$MSE(f,g) = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}(f_{ij} - g_{ij})^2 \tag{5.2}$$

Where:

f = reference image

g = test image

MxN = size of the image.

When the value of MSE reaches zero, which shows there is no difference between the reference image and test image, PSNR goes to infinity so higher values of PSNR indicate higher degree of similarities between the reference and test images. And on the other hand lower value PSNR indicates lower degree of similarities between the reference and test images.

5.2.2 SSIM

SSIM (structural similarity index measure) [39] is another widely used objective image quality evaluation metrics. It measure the similarity between two images by considering the structural degradation, luminance distortion and contrast distortion. The formula to calculate SSIM is given in Expression 5.3.

$$SSIM(f,g) = l(f,g)c(f,g)s(f,g) \tag{5.3}$$

Where l, c and s are measurements for luminance distortion, contrast distortion and structural degradation and are given by the following Expression 5.4.

$$\begin{aligned} l(f,g) &= \frac{2\eta_f\eta_g + c1}{\eta_f^2 + \eta_g^2 + c1} \\ c(f,g) &= \frac{2\sigma_f\sigma_g + c2}{\sigma_f^2 + \sigma_g^2 + c2} \\ s(f,g) &= \frac{\sigma_{fg} + c3}{\sigma_f\sigma_g + c3} \end{aligned} \tag{5.4}$$

where:

η_f = mean luminance of image f;

η_g = mean luminance of image g;

σ_f = standard deviation of image f;

σ_g = standard deviation of image g;

σ_{fg} = covariance between image f and g and

$c1$,$c2$ and $c3$ are constants added to avoid division by zero.

SSIM is in a range between [-1, 1]. In which 1 indicates higher degree similarity, 0 indicates lower degree of similarity and -1 indicates inverse correlation between the reference and test images.

5.2.3 Bjøntegaard Delta (BD)

Bjøntegaard Delta (BD) [40] is an evaluation metrics used to compare two different video codecs. It has two variations namely BD-PSNR, which calculates the quality difference between the two codecs at the same bitrate and BD-Rate, which calculates the bitrate reduction between the two codecs at the same quality. These metrics are calculated from

the rate distortion (RD) curve, which is a curve that tells us the quality values at different bitrates.

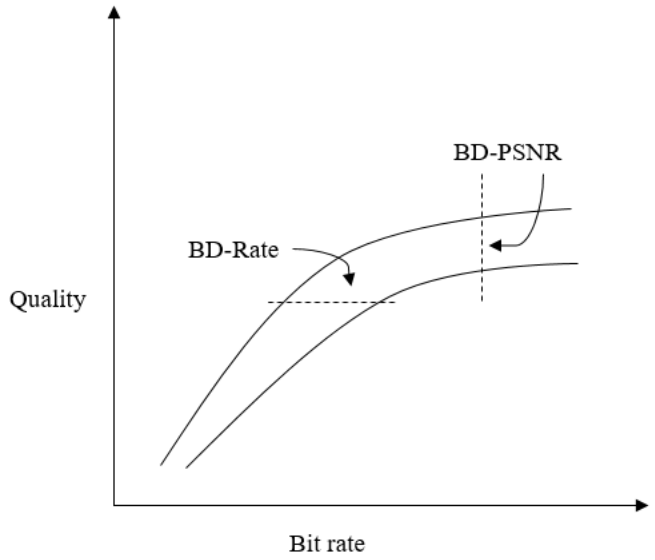

Figure 5.1: Example RD curve

5.3 Results

5.3.1 Experiment 1

RQ1. Does the temporal correlation that exist between consecutive frames in a video improve the quality of the reconstructed frame in CNN based bit depth enhancement algorithms?

The proposed model was compared with EBDA-CNN on its ability to restore the bit depth of the non-key frames. They were compared on the test dataset using the PSNR quality metric. Table 5.3 shows the bit depth enhancement performance of both models. It shows the average PSNR values of the two models for the test dataset. From the result we can see that the proposed model achieved PSNR improvement compared to the EBDA-CNN with an average PSNR of +1.6 dB. This shows that considering and exploiting the temporal correlation that exists in a video sequence improves the quality of the restored frame in CNN based bit depth enhancement algorithms.

Model	Average PSNR
EBDA-CNN	32.85
Proposed	34.45

Table 5.3: Average PSNR of the proposed approach and EBDA-CNN over the test dataset

5.3.2 Experiment 2

RQ2. Can we achieve more bitrate reduction by using this proposed bit depth enhancement CNN model in video coding schemes which are based on bit depth down sampling?

To answer this question we need to integrate the models in to the video coding scheme and test their performance. To compare their performance we need to see which model has better PSNR at the comparable video size by tuning the QP (quantization parameter). The key frames were compressed separately with different quantization parameter (an offset of 18 QP was added to the base QP to compress the key frames), because they had huge impact on the size of the compressed video. Table 5.4 and 5.5 shows the change (delta) in PSNR and SSIM between the proposed model and the EBDA-CNN on the test video sequences for different values of N, where N is equal to the number of frames between two key frames, and the base QP is set to 22. From the result it can be observed that the proposed model performs better than the EBDA-CNN especially for small value of N and when N increases it starts to perform worse for some cases.

The proposed approach performs better when the value of N is 5 with an average PSNR improvement of 0.648125, and it shows improvement for all the video sequences tested. The average PSNR improvement starts to decrease when the value of N increases and starts to perform worse for some video sequences when it reaches 20. At N is equal to 20 the proposed approach perform worse for the two video sequences PeopleOnStreet and Kimono1, it performs worse for four video sequences at N = 30 and for half of the video sequences at N = 50.

Among the test video sequences, the proposed approach achieved the highest PSNR improvement for the video sequences Johnny, KristenAndSara, BasketballDrill and FourPeople with an improvement of 1.79, 1.47, 1.04 and 0.92 respectively for N = 5, this is also true for the all N values. We believe the proposed approach achieved better improvement for these video sequences is because these videos have strong correlation within the frames of the video sequences. PeopleOnStreet and Kimono1 are among the video sequences that the proposed approach achieved the lowest PSNR improvement.

Figures 5.2 to 5.5 show sample RD curves of the proposed model and the EBDA-CNN for the

Video sequence	N				
	5	10	20	30	50
1	+1.04	+1.01	+0.9	+0.83	+0.87
2	+0.34	+0.22	+0.13	+0.08	+0.06
3	+0.53	+0.31	+0.07	-0.01	-0.1
4	+0.4	+0.22	+0.09	+0.05	-0.02
5	+0.67	+0.64	+0.64	+0.59	+0.62
6	+0.2	+0.08	-0.02	-0.07	-0.06
7	+0.45	+0.26	+0.07	-0.06	-0.16
8	+0.5	+0.37	+0.24	+0.18	+0.12
9	+0.58	+0.42	+0.2	+0.11	-0.01
10	+0.42	+0.21	+0.08	+0.04	-0.02
11	+0.31	+0.18	+0.09	+0.02	-0.03
12	+0.92	+0.95	+0.94	+0.96	+0.91
13	+1.79	+1.92	+1.84	+1.8	+1.72
14	+1.47	+1.49	+1.38	+1.31	+1.2
15	+0.25	+0.06	-0.05	-0.1	-0.15
16	+0.5	+0.48	+0.43	+0.4	+0.39
Average	+0.648125	+0.55125	+0.439375	+0.383125	+0.33375

Table 5.4: PSNR gain

Video sequence	N				
	5	10	20	30	50
1	+0.05	+0.05	+0.04	+0.04	+0.04
2	+0.01	0	0	-0.01	-0.01
3	+0.02	0	0	-0.02	-0.02
4	+0.02	+0.01	0	0	-0.01
5	+0.02	+0.02	+0.02	+0.02	+0.02
6	0	0	0	0	0
7	+0.02	+0.01	0	-0.01	-0.01
8	+0.02	+0.01	0	0	-0.01
9	+0.02	+0.02	0	0	0
10	+0.02	+0.02	+0.01	+0.01	0
11	0	0	-0.01	-0.01	-0.01
12	+0.03	+0.03	+0.03	+0.03	+0.03
13	+0.03	+0.03	+0.03	+0.03	+0.03
14	+0.03	+0.03	+0.03	+0.03	+0.03
15	+0.02	+0.02	+0.02	+0.01	+0.01
16	+0.02	+0.02	+0.02	+0.03	+0.03
Average	+0.020625	+0.016875	+0.011875	+0.009375	+0.0075

Table 5.5: SSIM gain

test video sequences by changing QP, for base QP (14, 18, 22, 26, and 30). The value of N is fixed at 5 for this experiment. From the figures it can be observed that the proposed model has an all over better coding gain than the EBDA-CNN. The figures show that the proposed approach has consistent coding gain for almost all the video sequences, except for Kimono1 (Figure 5.4) and PeopleOnStreet (Figure 5.5), over the different QP values. For these video sequences the proposed approach performs worse for QP 14. However the proposed approach achieved a better all over coding gain than the EBDA-CNN for all the video sequences.

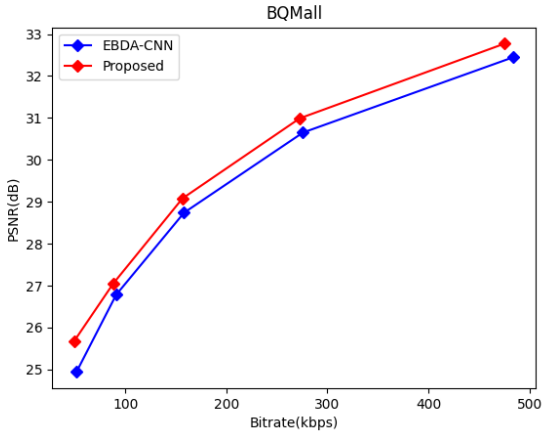

Figure 5.2: RD curve for BQMall

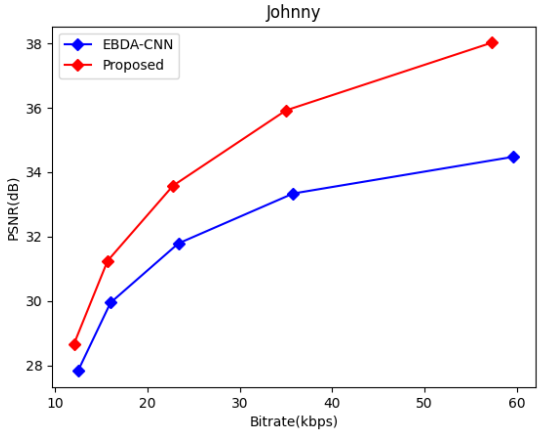

Figure 5.3: RD curve for Johnny

Table 5.6 shows the BD [40] (Bjøntegaard Delta (BD) measurement) results for rate and PSNR by comparing the two models. The proposed model achieved an average of -18.7454% BD-Rate gain over EBDA-CNN, this indicates that the proposed approach can produce same quality as the EBDA-CNN with 18% less bitrate (size). However the actual savings range between (-5.44295% to -36.7784%) depending on the content type. From the result it can

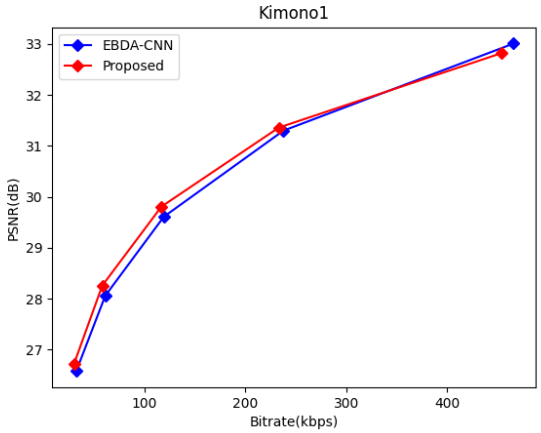

Figure 5.4: RD curve for Kimono1

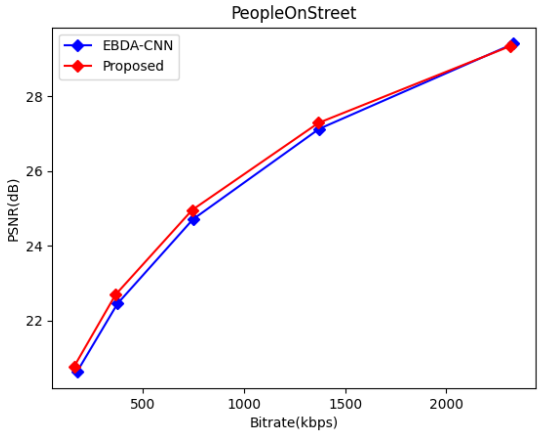

Figure 5.5: RD curve for PeopleOnStreet

be observed that the proposed approach achieved a BD-Rate gain over EBDA-CNN for all the test video sequences with video sequences Johnny, KristenAndSara, BasketballDrill and FourPeople being among the top ones with a gain of -36.7784%, -34.7981%, -26.3385% and -26.3385% respectively and PeopleOnStreet and Kimono1 being the least ones with a gain of -5.44295% and -6.56246% respectively.

5.4 Discussion

The conducted experiments over the test data sets and video sequences show that the proposed approach has a potential to improve the performance of video coding schemes that are based on bit depth down sampling. While doing these experiments we came up with three key findings.

Video Sequence	BD-PSNR	BD-Rate
1	+1.150054	-26.3385
2	+0.377638	-10.5214
3	+0.621447	-20.9485
4	+0.417796	-12.9396
5	+0.763365	-22.744
6	+0.155804	-6.56246
7	+0.458415	-18.8349
8	+0.584702	-15.5886
9	+0.688704	-22.0929
10	+0.406101	-12.4738
11	+0.297886	-10.6166
12	+1.217066	-25.256
13	+2.425443	-36.7784
14	+1.931354	-34.7981
15	+0.208586	-5.44295
16	+0.673172	-17.9905
Average	+0.773596	-18.7454

Table 5.6: BD-PSNR and BD-Rate of the proposed approach over EBDA-CNN the test video sequences

First, considering the temporal correlation that exists in consecutive frames of a video sequence and use previous frames which doesn't have their bit depth reduced (key frames) in restoring the frames which have their bit depth reduced (non-key frames) improved the quality of restored non-key frames. The second finding shows that the interval in which key frames are selected has an impact on the performance of the proposed approach. Finally integrating this bit depth enhancement technique in video coding schemes that are based on bit depth down sampling has a potential to improve the coding performance of these schemes.

In restoring the non-key frames which have their bit depth reduced the proposed bit depth enhancement CNN model achieved a better performance than the previous method with an average of +1.6dB improvement in PSNR. Based on the experiments on the impact of the interval between two key frames we observed that when the interval increases the proposed approach starts to perform worse for some of the videos. Regarding the coding performance the proposed approach, which takes in to consideration the temporal correlation that exists between consecutive frames of a video sequence achieved an average of -18.7454% BD-Rate gain over EBDA-CNN, which process each frame individually.

Chapter 6

Conclusion and recommendation

6.1 Conclusion

A raw video takes a lot of storage space and network bandwidth. Even though storage capacity and communication technologies improved over the years we still need to shrink the size of the video to efficiently use them. Video compression algorithms are used to reduce the size the videos and many of them have been proposed over the years to get a better bitrate reduction. Down sampling (in spatial resolution or bit depth) prior to encoding and up sampling it to the original form after decoding can be used to further reduce the bitrate in video coding.

In this paper a video coding scheme based on bit depth down sampling has been proposed. Unlike previous approaches the proposed approach exploits the temporal correlation that exists between consecutive frames of a video. It divides the frames into key and non-key frames. The key frames will be encoded without their bit depth reduced and will be used in restoring the bit depth of the non-key frames. Non-key frames will be encoded after their bit reduced and will be reconstructed at the decoder using a deep Convolutional Neural Network (CNN) which takes the non-key frame and a key frame as input. The results show that the proposed approach has a better performance in restoring the frames which have their bit reduced with an average of +1.6db improvement in PSNR and achieved better coding gain than the previous approach, which takes only the non-key frame as input to reconstruct it, with an average BD-Rate -18.7454%.

From the results of the experiments we conclude that considering the temporal correlation that exists in consecutive frames of a video sequence improves the performance of the bit depth enhancement techniques in restoring the frames which have their bit depth reduced

and the coding performance of video coding schemes that are based on bit depth down sampling can be improved by integrating this bit depth enhancement technique into the scheme.

6.2 Recommendation

The results from our study show that considering the temporal correlation exist between consecutive frames of a video sequence improve the performance of video coding schemes that are based on bit depth down sampling. However this study doesn't consider the motion exist between the frames and we believe the performance can be further improved by adding motion compensation to the proposed approach. Further studies can also be done to test the application of the proposed approach for high dynamic range contents and for different number of bit reductions.

- We will conduct a research by adding motion compensation to the bit depth enhancement CNN to see if it can be improved further.
- We will do research on the application of the proposed approach for high dynamic range contents (which have bit depth of 10, 12 or 16).

Bibliography

[1] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra. Overview of the h.264/avc video coding standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(7):560–576, 2003.

[2] G. J. Sullivan, J. Ohm, W. Han, and T. Wiegand. Overview of the high efficiency video coding (hevc) standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 22(12):1649–1668, 2012.

[3] C. C. Aggarwal. "convolutional neural networks" in neural networking deep learning: A textbook. pages 315–371, 2018.

[4] W. Cui, T. Zhang, S. Zhang, F. Jiang, W. Zuo, Z. Wan, and D. Zhao. Convolutional neural networks based intra prediction for hevc. In *2017 Data Compression Conference (DCC)*, pages 436–436, 2017.

[5] Y. Wang, X. Fan, S. Liu, D. Zhao, and W. Gao. Multi-scale convolutional neural network-based intra prediction for video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(7):1803–1815, 2020.

[6] T. Dumas, A. Roumy, and C. Guillemot. Context-adaptive neural network-based prediction for image compression. *IEEE Transactions on Image Processing*, 29:679–693, 2020.

[7] T. Laude and J. Ostermann. Deep learning-based intra prediction mode decision for hevc. In *2016 Picture Coding Symposium (PCS)*, pages 1–5, 2016.

[8] N. Song, Z. Liu, X. Ji, and D. Wang. Cnn oriented fast pu mode decision for hevc hardwired intra encoder. In *2017 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pages 239–243, 2017.

[9] Z. Liu, X. Yu, Y. Gao, S. Chen, X. Ji, and D. Wang. Cu partition mode decision for hevc hardwired intra encoder using convolution neural network. *IEEE Transactions on Image Processing*, 25(11):5088–5103, 2016.

[10] N. Yan, D. Liu, H. Li, and F. Wu. A convolutional neural network approach for half-pel interpolation in video coding. In *2017 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–4, 2017.

[11] H. Zhang, L. Li, L. Song, X. Yang, and Z. Li. Advanced cnn based motion compensation fractional interpolation. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 709–713, 2019.

[12] N. Yan, D. Liu, H. Li, B. Li, L. Li, and F. Wu. Convolutional neural network-based fractional-pixel motion compensation. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(3):840–853, 2019.

[13] Z. Zhao, S. Wang, S. Wang, X. Zhang, S. Ma, and J. Yang. Cnn-based bi-directional motion compensation for high efficiency video coding. In *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–4, 2018.

[14] S. Huo, D. Liu, F. Wu, and H. Li. Convolutional neural network-based motion compensation refinement for video coding. In *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–4, 2018.

[15] J. Lee, K. Kong, G. Bae, and W. J. Song. Blocknet: A deep neural network for block-based motion estimation using representative matching. *Symmetry*, 12:840, 05 2020.

[16] H. Choi and I. V. Bajić. Deep frame prediction for video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(7):1843–1855, 2020.

[17] J. K. Lee, N. Kim, S. Cho, and J. Kang. Convolution neural network based video coding technique using reference video synthesis. In *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pages 505–508, 2018.

[18] L. Zhao, S. Wang, X. Zhang, S. Wang, S. Ma, and W. Gao. Enhanced motion-compensated video coding with deep virtual reference frame generation. *IEEE Transactions on Image Processing*, 28(10):4832–4844, 2019.

[19] D. Liu, H. Ma, and Z. Xiong. *CNN-Based DCT-Like Transform for Image Compression*, pages 61–72. 01 2018.

[20] M. Alam, T. Nguyen, M. Hagan, and D. Chandler. A perceptual quantization strategy for hevc based on a convolutional neural network trained on natural images. page 959918, 09 2015.

[21] C. Ma, D. Liu, X. Peng, L. Li, and F. Wu. Convolutional neural network-based arithmetic coding for hevc intra-predicted residues. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(7):1901–1916, 2020.

[22] T. Chen, H. Liu, Q. Shen, T. Yue, X. Cao, and Z. Ma. Deepcoder: A deep neural network based video compression. In *2017 IEEE Visual Communications and Image Processing (VCIP)*, pages 1–4, 2017.

[23] C.-Y. Wu, N. Singhal, and P. Krähenbühl. Video compression through image interpolation. In *ECCV*, 2018.

[24] G. Lu, W. Ouyang, D. Xu, X. Zhang, C. Cai, and Z. Gao. Dvc: An end-to-end deep video compression framework. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10998–11007, 2019.

[25] Z. Chen, T. He, X. Jin, and F. Wu. Learning for video compression. *IEEE Transactions on Circuits and Systems for Video Technology*, 30:566–576, 2020.

[26] Y. Su, W. Sun, J. Liu, G. Zhai, and P. Jing. Photo-realistic image bit-depth enhancement via residual transposed convolutional neural network. *Neurocomputing*, 347, 04 2019.

[27] J. Liu, W. Sun, Y. Su, P. Jing, and X. Yang. Be-calf: Bit-depth enhancement by concatenating all level features of dnn. *IEEE Transactions on Image Processing*, 28(10):4926–4940, 2019.

[28] C. Peng, L. Cai, Z. Fu, and X. Li. Cnn-based bit-depth enhancement by the suppression of false contour and color distortion. In *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pages 1145–1151, 2019.

[29] Y. Zhao, R. Wang, W. Jia, W. Zuo, X. Liu, and W. Gao. Deep reconstruction of least significant bits for bit-depth expansion. *IEEE Transactions on Image Processing*, 28(6):2847–2859, 2019.

[30] J. Liu, P. Liu, Y. Su, P. Jing, and X. Yang. Spatiotemporal symmetric convolutional neural network for video bit-depth enhancement. *IEEE Transactions on Multimedia*, 21(9):2397–2406, 2019.

[31] V. Nguyen, D. Min, and M. N. Do. Efficient techniques for depth video compression using weighted mode filtering. *IEEE Transactions on Circuits and Systems for Video Technology*, 23(2):189–202, 2013.

[32] F. Zhang, M. Afonso, and D. R. Bull. Enhanced video compression based on effective bit depth adaptation. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 1720–1724, 2019.

[33] D. Ma, F. Zhang, and D. R. Bull. Gan-based effective bit depth adaptation for perceptual video compression. In *2020 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6, 2020.

[34] Yuv21. `http://www.codersvoice.com/a/webbase/video/08/152014/130.html`. Accessed: 2021-07-03.

[35] harmonic. `https://www.harmonicinc.com/free-4k-demo-footage/`. Accessed: 2021-07-03.

[36] Uvg. `http://ultravideo.cs.tut./`. Accessed: 2021-07-03.

[37] xphi. `https://media.xiph.org/video/derf/`. Accessed: 2021-07-03.

[38] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 12 2014.

[39] A. Horé and D. Ziou. Image quality metrics: Psnr vs. ssim. In *2010 20th International Conference on Pattern Recognition*, pages 2366–2369, 2010.

[40] G. Bjontegaard. Calculation of average psnr differences between rd-curves. 2001.