Implementation of Novel Image Compression Techniques using Machine Learning Algorithms on HEVC

Submitted in partial fulfillment of the requirements for the award of the degree of

DOCTOR OF PHILOSOPHY

by

KANIKE SREENIVASULU

(Roll No. 717035)

Under the Supervision of
Prof. T V K Hanumanth Rao

Professor

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY
WARANGAL – 506004, T.S, INDIA

November – 2022

APPROVAL SHEET

This thesis entitled "**Implementation of Novel Image Compression Techniques using Machine Learning Algorithms on HEVC**" by **Mr. Kanike Sreenivasulu** is approved for the degree of **Doctor of Philosophy**.

Examiners

____

____

Supervisor

____

Prof. T V K HANUMANTH RAO

Professor, Electronics and Communication Engineering Department,

NIT WARANGAL

Chairman

____

Dr. P. Sreehari Rao

Head, Electronics and Communication Engineering Department,

NIT WARANGAL

Date:

Place:

DECLARATION

I, hereby, declare that the matter embodied in this thesis entitled "**Implementation of Novel Image Compression Technique using Machine Learning Algorithms on HEVC"** is based entirely on the results of the investigations and research work carried out by me under the supervision of **Prof. T V K Hanumanth Rao**, Department of Electronics and Communication Engineering, National Institute of Technology Warangal. I declare that this work is original and has not been submitted in part or full for any degree or diploma to this or any other University.

I declare that this written submission represents my ideas in my own words and where other ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated, or falsified any idea/date/fact/source in my submission. I understand that any violation of the above will cause disciplinary action by the institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Kanike Sreenivasulu

Roll No: 717035

Date:

Place: Warangal

Department of Electronics and Communication Engineering
National Institute of Technology
Warangal – 506 004, Telangana, India

CERTIFICATE

This is to certify that the dissertation work entitled **"Implementation of Novel Image Compression Techniques using Machine Learning Algorithms on HEVC,"** submitted by Mr. Kanike Sreenivasulu (Roll No. 717035), is a bonafide work submitted to the National Institute of Technology Warangal in partial fulfillment of the requirement for the award of the degree of Doctor of Philosophy to the Department of Electronics and Communication Engineering of National Institute of Technology Warangal, is a record of bonafide research work carried out by him under my supervision and has not been submitted elsewhere for any degree.

Prof. T V K Hanumanth Rao
(Supervisor)
Professor, Department of ECE
National Institute of Technology
Warangal, India – 506004

ACKNOWLEDGEMENTS

I would like to thank a number of people who have contributed to my Ph.D. directly or indirectly and in different ways through their help, support, and encouragement.

Firstly, I would like to express my sincere gratitude to my supervisor **Prof. T V K Hanumanth Rao,** for his patience, motivation, and guidance. My sincere thanks to him for providing me with an opportunity to join the institute as a Ph.D. research scholar and giving me access to research facilities. Without the precious support, it would not have been possible to conduct this research.

I thank the current Head of The Department, **Dr. P. Sreehari Rao**, and the former Heads, **Prof. T. Kishore Kumar, Prof. N. Bheema Rao** and **Prof. L. Anjaneyulu** for providing all the necessary support from the department.

Besides my supervisor, I would like to thank the rest of my Doctoral Scrutiny Committee members**: Prof. T. Kishore Kumar, Prof. N. Bheema Rao, Dr. P. Sreehari Rao** and **Dr. P. Muthu,** for their insightful comments and encouragement, but also for the hard questions which incented me to widen my research from various perspectives.

I thank **Prof. N.V. Ramana Rao**, Director, NIT Warangal, for providing official support and financial assistance to carry out the research.

I thank my past and present fellow at NIT Warangal with, whom I have had the pleasure to work with.

KANIKE SREENIVASULU

ABSTRACT

Conventional High Efficiency Video Coding intra and inter frame prediction techniques depend on similar features within the frame and between the frame. Since Intra and inter frame predictions through prediction calculations related to current block, its adjoining blocks and reference frame, which decrease the quantity of prediction mode search during the coding tree unit's prediction. The conventional hevc video processing algorithms lack multi-level optimization and the coding unit size of High Efficiency Video Coding is larger than 16×16 blocks which increases computational complexity of the encoder. In addition, the prediction of the coding tree unit block region in the near frame is linked to adjacent samples inside a frame and the pixels of the border region, which introduce redundancy and noise which ultimately decrease quality of frame at the receiving end.

To overcome this issue, a convolutional neural network (CNN) based sequential ensemble learning technique and deep neural network long short term memory (LSTM) is proposed for intra and inter prediction for High Efficiency Video Coding, which ensemble generation to improve the prediction of enormous motion varieties and provide optimal reconstruction through prediction of spatial adjoining frames. For High Efficiency Video Coding intra and inter frame prediction, the proposed approach effectively accomplishes better video coding quality by consolidating the predictions through convolutional neural network (CNN) ensemble and long short term memory (LSTM) process. The exploratory outcomes shows that the proposed convolutional neural network (CNN) based intra and long short term memory (LSTM) based inter prediction through sequential ensemble learning technique show excellent frame prediction and quality of reconstructed frames in terms of peak signal to noise ratio and structural similarity index compared to existing intra and inter prediction techniques.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBRIVATIONS

VRCNN- Variable-Filter-Size Residue-Learning

CNN-Convolutional Neural Network

LSTM-Long Short Term Memory

JPEG-Joint Photographic Expert Group

MJPEG-Moving Picture Joint Photographic Expert Group

DS-CNN- Decoded Side Convolutional Neural Network

QE-CNN-Quality Enhancement Convolutional Neural Network

TQEO- Time-constrained Quality Enhancement Optimization

MSDD- Multi-Scale Deep Decoder.

BD-rate- The Bjontegaard rate difference

CABAC- Context-adaptive binary arithmetic coding

PSNR- Peak Signal to Noise Ratio

SSIM-Structural Similarity Index

IP-Internet Protocol

DWT-Discrete Wavelet Transform

IDWT-Inverse Discrete Wavelet Transform

SPIHT-Set Partitioning in Hierarchical Tree

MSE-Mean Square Error

HSV Image- Hue Saturation Value Image

HEVC-High Efficiency Video Codec

CTU-Coding Tree Unit

CB-Coding Block

PU-Prediction Unit

MPEG-Moving Picture Expert Group

AVC-Advanced Video Codec

SSE- The Sum of Square Error

RDO- Rate-Distribution Optimization

GoP-Group of Pixels

RNN-Recurrent Neural Network

CONTENTS

CHAPTER-1

INTRODUCTION

1.1 Introduction

Video Compression Algorithms are algorithms that alter video signals to reduce the amount of storage and bandwidth needed. For creators of embedded systems, processors, and tools that target video applications, understanding how video codecs function is critical.

A video clip is made of a series of distinct images known as frames. As a result, many of the concepts and approaches used in still-image compression algorithms, such as JPEG, are also used in video compression algorithms. One method for compressing video is to remove similarities between subsequent video frames and compress each frame independently of others. Employing the JPEG still-image compression standard to compress video streams is an example. Motion JPEG, or MJPEG, is a method that is sometimes used to create new video applications. Although recent video compression algorithms go beyond still-image reduction schemes and take advantage of similarities between consecutive video frames utilising Motion Estimation and Motion Compensation, they nevertheless leverage principles from still-image compression algorithms. As a result, we begin our investigation of video compression by exploring the inner workings of transform-based still-image compression techniques such as JPEG, as well as applying various Deep Neural Network and Machine Learning Algorithms to video codecs like HEVC.

Nowadays, digital video has become an essential tool in many ways. Both video transmission and teleconferencing have ensured that digital video is a global phenomenon for industries as well as the common man. To cite an example, an increasing growth in video sharing services, such as YouTube and Netflix demonstrate the major function that this medium perform in our day-to-day activities. Creation of mobile video and the attractiveness of high definition (HD), ultrahigh definition (UHD) video content have made video data the most often shared form of content across transmission networks worldwide.

High-resolution, high-frame rate video contents require extreme bitrates that are unrealistic to be accommodated in modern transmission structures, particularly in wireless transmission networks. Therefore, the arena of video compression and transmission are regularly taken up by both academic and industry-oriented researches on discovering processes for further improving efficiency in compression quality. Thus, the process of compression in videos has been evolving constantly from the early 1990s to provide updated requirements. Over the past decades, the H.624/AVC (Advance Video Coding) standard was introduced as a well-known video compression standard. However, the H.264/AVC's compression efficiency has been inadequate in meeting the demands of the exponentially increasing video traffic. Hence, the constant upgrade in video compression is vital to deal with the ever-increasing requirements of the visual media. In earlier 2013, the emergence of a next-generation video codec, i.e., HEVC/H.265(High Efficiency Video Coding) was designed for providing future high-resolution video contents and effectively utilizing the recent parallel processing infrastructures in modern general-purpose processor (GPP) including the digital signal processor (DSP). The Joint Collaborative Team on Video Coding (JCT-VC) society, a collaboration of two prominent standardization organizations, such as the International Standards Organization/International Electrotechnical Commission (ISO/IEC) Moving Pictures Expert Groups (MPEG) and ITU Telecommunication Standardization Sector (ITU-T) Video Coding Experts Group (VCEG), has developed the HEVC. Now, it is estimated that HEVC has been instrumental for achieving half of half compression effectiveness when associated with H.264/AVC.

It is now very much easier to compress the video sequences much more professionally than any other conventional standards using HEVC. Further, it offers more flexible options while applying it to a wide range of networks. For achieving efficient compression, the cost involved is high. In spite of this, such desired compression efficiencies are impossible to achieve across various domains of the visual media. Devices, such as mobile phones and other embedded systems, exploit simple encoders or simpler profiles of the codec to trade off efficiency in compression and quality.

Currently, however, organizations use visual media based on various video coding standards designed by numerous standardizing units. Despite improvements in compression efficiency, HEVC presents several challenges with respect to its use, keeping in mind with limited processing and energy resources due to its coding requirements and characteristics. Thus, an improvement in inter and intra prediction is needed to further increase both coding and compression efficiency sufficiently with a reduced number of storage demands.

1.1.1 Types of Image Compression

Two different kinds of image compression methods are in vogue:.

1. Lossy image compression
2. Lossless Image compression

1.1.2 Lossy Image Compression

Higher levels of data reduction are achieved with lossy compression, but the original image is not perfectly replicated. High compression ratio is offered. In applications like broadcast television, videoconferencing, and facsimile transmission, where a certain degree of inaccuracy is a reasonable trade-off for improved compression efficiency, lossy image compression is helpful.

1.1.3 Lossless Image Compression

The only acceptable level of data reduction is lossless image compression. Comparatively speaking of lossy, it offers low compression ratio. Lossless image compression approaches are made up of two largely independent steps: (1) creating a different representation of the image with fewer redundant pixels, and (2) coding the representation to remove redundant codes. Applications including corporate papers, satellite photos, and medical imaging all benefit from lossless image compression.

1.1.4 Image Compression Process

Various image compression techniques convert the target image into the frequency domain. The altered images are finally transformed at the receiving end after being encoded using various encoding methods. At the receiving end, the bit streams are then obtained. The retrieved data is decrypted. In the end, an inverse transform is performed to produce a compressed image.

1.1.5 Necessity of Image Compression Techniques

Coding redundancy, interpixel redundancy, and phychovisal redundancy are the three fundamental data redundancies that can be found and used in picture compression. When one or more of these redundancies exist, image compression is utilised to reduce them.

The main applications for picture compression are image transmission and storage. Applications for image transmission include broadcast television, teleconferencing, computer communications, remote sensing via satellite, aircraft, radar, or sonar, and facsimile transfer. Documents used in education and business, medical images produced by computer tomography

(CT), magnetic resonance imaging (MRI), and digital radiology, movies, satellite photos, weather maps, geological surveys, and other types of media all require image storage in one form or other.

1.1.6 Image Compression Methods

Data compression, which uses fewer bits to encode the actual image, is where image compression first emerged. Any image compression technique's primary goal is to minimise the image's storage footprint. It is categorised as lossy or lossless depending on the quality requirement. In a lossless scenario, there is no data loss because the decompressed image is identical to original image. Statistical techniques including Huffman coding, Run Length coding, Arithmetic coding, and dictionary-based methods like Lempel-Ziv-Welch (LZW) coding, and these are categorised as lossless methods. The image compression literature also contains hybrid versions of these techniques.

However, because the lossy approach is irreversible, data loss is a possibility. It is primarily categorised as methods using the spatial and frequency domains. Only the spatial elements of the image are considered and further processed in spatial domain approaches. It consists of Block Truncation Coding (BTC) and Vector Quantization. Another lossy compression technique that uses a fractal dimension is called fractal coding. Since the computation is much simpler with the frequency component, frequency domain algorithms totally convert the image to the frequency domain. Transformation can be done by means of various transforms such as Fourier Transform, Singular Value Decomposition (SVD) based methods, Karhunen-Loeve Transform (KLT), Discrete Cosine Transform (DCT) and Wavelet Transform.

Fourier transform is a well-known method for processing signals and images; however, it only provides information on the frequencies that appear in a signal and not their timing. In other words, time-frequency analysis is not committed, and it is only appropriate for stationary signals. However, many of the images, including those in the medical and natural sciences, are not stationary in nature. Numerous time-frequency analysis techniques have been proposed, such as Short-Term Fourier Transform (STFT), Wigner-Ville Distribution Function (WDF), Hilbert-Huang Transform (HHT), and Discrete Cosine Transform (DCT). These transforms have some limitations including size of the window and fixed resolution analysis in STFT with only partial time domain details in DCT and cross term problem with multi-component images in WDF. Unlike all other transforms, wavelets and SVD provide perfect reconstruction with multi-level decomposition and reduces blocking artifacts.

In order to achieve better compression performance and good image reconstruction quality, near lossless approaches were developed. Joint Photographic Expert Group (JPEG) is one of the compression standards that can be both lossless and lossy. Lossless version of JPEG is indicated as JPEG-LS (LOCO-l). The advancement of JPEG is implemented as JPEG2000 which incorporates wavelet and the Embedded Block Coding with Optimized Truncation (EBCOT). It comes with many variants such as ROI coding, high fidelity ratio coding and JPEG2000 for high dimensional data. However, these lossy/lossless/near-lossless compression techniques may be embedded with several object-based image coding techniques for efficient compression of medical images.

The premise behind wavelet coding is that the transforms co-efficient, which de-correlates an image's pixels, can be coded more effectively than original pixels themselves. The transform's basis functions, in this case wavelets, allow the majority of the essential visual information to be condensed into a small number of co-efficients, which enable the remaining co-efficient to be coarsely quantized or truncated to zero with little to no image distortion.

Due to the wavelet transforms' energy compaction properties and multi resolution properties, as well as their capacity to handle signals, modern discrete wavelet based coders have outperformed discrete cosine transform-based coders in terms of still image compression, offering higher compression ratios and more peak signal to noise ratios (PSNR).

1.2 Video Sequence

A video sequence has a set of image frames depicted one after another in sequence. Each frame has a set of pixels, and each pixel holds information, such as the intensity of the color to be shown. An image frame represented by RGB (Red, Green and Blue) color space will have intensity values for each RGB channel.

1.2.1 Frames per Second

In order to display a video smoothly, the rate of the depicted image sequence requires 24–30 frames per second, which have been the benchmark in television and movie industry for a long time. New video formats maintain frame rates beyond 30 frames per second for increasing the perceived smoothness of motion. Because the frame rate and the resolution of the video increases, the image level information stored in the video sequence also increases. A raw 4K video at 60 fps holds eight times the image information of a raw high definition (HD) at 30fps.

1.2.2 Video Color Space

The colors of an image may be denoted by using standard color components RGB(red, green and blue). Individual pigment component is allocated an intensity value to be shown by devices using the RGB color space, e.g., computer and television monitors. Typically, video compression and communication use a compressed color space termed YCbCr or YUV represented by one light intensity component i.e., luminosity (Y) and two color components i.e., chromatic blue (Cb or U) and chromatic red (Cr or V). The YUV color space can convert RGB colors to fewer bits of color and still seem attractive, much similar to the human eye. The mapping between the RGB and YUV color spaces may be expressed by matrix multiplication.

The mapping matrix coefficients vary between standards however, the International Telecommunication Union (ITU) has come up with released recommendations for both standard and HD Television (TV). The constraints of human vision may be further developed by compressing YUV chromatic components using color chroma subsampling (SS).

1.3 Video Coding

It's a compression method used for discovering unwanted information in a video sequence and compressing it efficiently to achieve smaller video file sizes resulting in fewer bits in the video bit stream. This method comprises encoding and decoding, where the encoder compresses the video and encodes it into a bit stream, which is received by the decoder for reconstructing the compressed video sequence.

1.3.1 Video Coding Standard

The modern H.26X standards are grounded on the principles of H.261 video coding standard released earlier. H.261 introduces a new way of video coding; namely, hybrid video coding, where decoding techniques are involved in the encoding process that facilitate for effective prediction modes (PMs), so that only the difference between interframes need to be transmitted. Since H.261, each new H.26X standard builds on the previous one and makes proper advances for the requirements of the present video qualities and resolutions.

Currently, AVC/H.264 remains one of the most broadly utilized video coding norms in HD television content, Blu-ray and HD streaming services, such as YouTube and Netflix. Even though AVC was developed for standard definition television, it has been used mostly for HD content. AVC was under progress from 1999 until 2003 and extended between 2003 and 2009. The successor of AVC is HEVC/H.265 standard. It tries to increase the compression efficiency for HD content, while maintaining high quality. HEVC was developed with Ultra HD (UHD), e.g., 4K and 8K resolutions. The objective of HEVC was adept to have the similar video resolution and perceived quality at half the bit rate when compared with AVC for HD content which was achieved before the release of standard model.

1.4 Video Quality Measure

The quality measures for video sequences may be measured directly or indirectly. The objective measurements are performed accurately using standardized measurements, where the pixel variances between a reference image and a compressed image are compared. The objective measurements can provide accurate measurements; however, it is also often required to perform the individual measurements for evaluating the perceived visual quality. Although an objective measurement points to degradation in video quality, the human eye cannot observe the degradation and so an individual quality possibly will be maintained at the same level.

The most frequent objective measurements include,peak signal-to- noise ratio (PSNR), SSIM and mean square error (MSE). It is familiar to compare PSNR between a baseline video coder and a suggested improvement to the video coder at different bitrates, called Bjontegaard delta (BD) rate.

1.4.1 Mean Square Error

The MSE is based on two straight forward mathematical formulae, such as SAD and SSE, all of which are used for comparing two images pixel by pixel. SAD, SSE and MSE are the most commonly used in video coding for computing the variance between images in the actual video sequence and the compressed video sequence. The SAD formula computes the absolute pixel value variances as shown in Equation (1.1). SSE acquires the formula one step further by squaring the pixel value variance as Equation (1.2). MSE obtains the SSE outcome and computes the mean pixel variance by splitting the entire pixels of the images that were compared in Equation (1.3). The MSE measurement may be directly converted into a PSNR.

The variable X is the decoded compressed image array and Y is the actual image array while, h = $height$, N = $width$ of the images and I= (X, Y).

$$(I) = \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} |[A] - X[A]| \quad \text{, whereas } A=[i,j] \qquad (1.1)$$

$$(I) = \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} (Y[A] - X[A])^2 \qquad (1.2)$$

$$\text{MSE(I)= SSE(I)/(M-N)} \qquad (1.3)$$

1.4.2 PSNR

It is employed for comparing the excellence of a reference image or sequence of images to the compressed version of the identical image or sequence of images. It is commonly used for measuring the effects of video compression. It is defined as the ratio between the actual image signal and the noise signal introduced using lossy compression. The maximum number of pixel values based on the number of bits (B) is squared and divided by the MSE of an image. PSNR is computed as follows:

$$\text{PSNR}=10.\log_{10}((2^B-1)^2)/\text{MSE}$$

It is measured in logarithmic unit decibel (dB). For lossy compression at a depth of 8 bits, the PSNR is normally between 30 and 50 dB. It is often used for comparing video codes; however, it must not be viewed as a representation of the perceived video quality as PSNR value may differ depending on the image or video content. PSNR is applied to the individual Y, Cb and Cr components. Typically, a total PSNR is weighed depending on the chroma SS.

1.4.3 Structural Similarity Index (SSIM)

SSIM is used for measuring the structural similarities between images and predicting the perceived quality. SSIM weighs in the structural distortions, such as noise, blocking artifacts, blurring and ringing, which are simply observed by the human eye. Although outcomes may vary between SSIM and PSNR, they often correlate in image compression conditions. Changes in luminance, contrast, gamma and SP shifts are nonstructural distortions and do not modify the structure of the image. The SSIM given in Equation (1.6) is built by three components that measure the variance in luminosity, contrast and structure.

If X and Y are images to be compared using SSIM, $(A,)$ compared the luminosity, (A, B) compares the contrast and (A, B) compares the structure. The three components can be weighted directly depending on the application; however, normally weighed equally as

$wl = wc = ws = 1$, R=(X, Y)

$$(R) = [(R)] \cdot [(R)] \cdot [(R)] \quad (1.6)$$

1.5 Motivation

Emerging technology has driven the conventional communication towards the better services provisioning with respect to accuracy and speed of transmission. Based on investigation into the efficacy of communication system, the demand for better coding technique and high rate services are increasing. Traditional communication architectures are capable of transmitting low rate services such as voice or text data. Current communication architecture is used for very high rate services such as video streaming, IP Services, Multimedia services, videoconferencing etc. As the rate of transmission increases, the transmission errors for end-to-end communication increases. In case of video coding, this error is very conspicuous, as the data representing the video information is very high. The progressive transmission of video data over a Communication channel corrupts information and to recover the corrupted signal back, estimation algorithms have been proposed which are complex in computation. Various video coding standards like H.264 or HEVC were proposed for coding video data but these architectures do not specify what action a decoder should take when an error is detected.

Exploitation of spatial and temporal redundancies in HEVC video data is one of the most important processes in video encoding procedures, contributing to the high compression capability of the H.265 architecture, one of the latest video codecs. Use of an LSTM based deep learning approach to carry out inter-frame prediction with a sequence of N previous frames to obtain a predicted (N+1) th frame, with which a residual frame is generated, occupying much less space when encoded is currently being done. With the presence of artifacts during the quantization process, frame distortions take place in HEVC by CNN this can be limited by training and testing components through CNN-based ensemble learning method, proposed in this paper. Coordinating the HEVC in-loop filters to intra-coding sampling filters reduces the resolution of the processed frame, using CNN this can be limited by utilizing the down-sampled block for encoding the processed frame through CNN-based learning block method. PSNR and SSIM are low in HEVC predicted Frame, in order to Improve it, CNN based ensemble algorithm and LSTM are proposed.

1.6 Problem Statement

HEVC, also known as H.265, is widely used for video processing. It performs intra and inter-prediction by comparing pixels and previous blocks, which results loss of valuable information when weak signal prediction occurs. To overcome this problem, intra and inter frame prediction using Convolutional Neural Network based Ensemble learning (different number of convolutional filters are applied to different images) and LSTM are proposed, which provide valuable context of blocks and ultimately improve efficiency of HEVC.

1.7 Objectives

- To study the architecture of HEVC/H.265 video coding algorithms.
- To identify the research gap and to develop novel image compression techniques based on HEVC standered.
- To increase efficiency of the existing HEVC coding algorithm.
- To develop and implement novel video compression techniques using inter frame and intra frame techniques of HEVC
- To compare the results of novel video compression algorithms with the existing algorithms of HEVC/H.265.

1.8 Organization of the Thesis

This thesis is organized in seven chapters

Chapter 1: This chapter discusses the rationale for the study the requirement for H.265, as well as previous research on H.265 to increase its performance. The thesis goals are also summarised in this chapter.

Chapter 2: This chapter discuses about various existing techniques and their limitation with regard to high efficiency video codec intra and inter frame prediction techniques.

Chapter 3: This chapter describes basics and different types of images. This chapter also describes About image compression techniques using seam carving and integer wavelet transform. Finally, it explains pros and cons and applications of discrete wavelet transform.

Chapter 4: This chapter describes basics and Architecture of HEVC.

Chapter 5: This chapter describes basics and working of CNN and Ensemble Learning Algorithms to predict frames of HEVC and also explains the advantages, drawbacks and applications of CNN.

Chapter 6: This chapter discusses the problems with RNN and ways to improve RNN; It also Describe the architecture of LSTMs. Finally, it elucidates results and discussions pertaining to LSTM for inter frame prediction of HEVC.

Chapter 7: This chapter concludes the study and makes recommendations for future research.

CHAPTER 2

LITERATURE SURVEY

In this chapter the related work for this research work and existing prediction techniques to improve HEVC/H.265 inter- and intra-coding are discussed.

2.1 Intra coding Techniques for HEVC

Cho & Kim (2013) suggested a rapid CU divide and trim for suboptimal CU segregation in HEVC intracoding that allows a drastic reduction in the computational complication with a minor degradation in RD act. This algorithm was performed based on two completing steps, such as primary CU split and trim decision. For CU blocks, the initial CU divide and trim tests were achieved at every CU depth level using Bayes decision manner on the basis of little difficulty RD costs and entire RD costs, respectively. However, the more the speedups of the CU splitting (CUS) and trim processes, the more will be the degradation in the coding efficiency because of increased misclassification in the divided and unsplit CU choice of the initial CUS and trim tests.

Pastuszak & Abramowski (2016) presented a computationally scalable algorithm and hardware architecture for intra-encoding. The suggested encoder was used to allow the deals among the “compression efficiency and throughput”. The pre-option-based prediction brings about the samples carried out with the same resources as routine processing. Furthermore, the hardware expenditure was brought down, and high throughputs were gained. Still, the power saving of this algorithm was limited and quality loss was soaring.

The sampling ratio is a block four pixels wide and two pixels high, and is shortened to the format 4: X: Y where X and Y denote the row- wise chroma sampling parameters in a 4×2 pixel block (Kerr 2012).

The result of chroma SS is a reduced image color resolution and the image quality is affected on a pixel level (Sullivan et al. 2012; Ohm et al. 2012); however, the human eye will hardly be able to observe it from a standard viewing distance due to it being more sensitive to light than color and spatial shift displacement produces higher error rate, which leads to poor video quality (Lee et al. 2014). Normally, 4: 2: 0 is used in video compression, and is implemented in both AVC and HEVC (Wien 2015). Typically, each sample is denoted by eight bits (0–255) of precision; however, ten bits (0–1024) are also used within the HEVC.

The YUV color space can convert RGB colors to fewer bits of color and still seem attractive and much similar to the human eye. The mapping between RGB and YUV color spaces may be expressed by matrix multiplication (Sector 2015).

HEVC was developed with Ultra HD (UHD), e.g., 4K and 8K resolution. The objective of HEVC was to have similar video resolution and perceived quality at half the bit rate when compared with AVC for HD content which was achieved before standardization (Tan et al. 2015). It performs coding on a pixel block basis and each image of the video sequence is divided into coding blocks to be encoded (Sze et al. 2014).

PSNR is often used for comparing video codes; however, it must not be viewed as a representation of the perceived video quality as PSNR value may differ depending on the image or video content (Huynh-Thu & Ghanbari 2008).

2.2 Intra Prediction Mode Decision Techniques in HEVC

Jain & Rao (2014) proposed a fast-intra-MD in HEVC that stops the complete full search prediction for the CU. This was then followed by a PU mode decision for finding the optimal modes HEVC encoder 35 PM. Initially, the SAD of all the modes was computed using downsampling method and then a three-step search algorithm was applied to eliminate any redundant modes. This was then followed by an early RDO quantization termination algorithm which will further reduce the encoding term. Then again, it requires an adaptive threshold (TH) to terminate the MD and further reduce encoding time.

Gan et al. (2015) proposed a procedure with premature termination of CU split and MD. In this algorithm, the variance of the input image was used, so as to end the CU parting early In addition, the adjacent mode was used in order to end early the RDO procedure depending on analysis of candidate modes (CM), which were obtained by rough MD so as to diminish the CC. However, a considerable reduction in time was then achieved by reducing the PSNR and thereby increasing the BR; i.e., PSNR was not increased that degrades the video quality.

Ma et al. (2018) initiated a fast intracoding algorithm built on the CU extent choice and DIR- MD for HEVC. Initially, an agility CU extent decision scheme was introduced to choose diverse depth choice approaches for every major coding unit. Then, a pace DIR- MD scheme was initiated. Initially, the DIR modes of the parent unit and the most probable modes (MPMs) lists were compared. Formerly, the primary DIR mode of RDO index was used for a premature termination of RDO processes. However, the TH was predetermined, such that the RD charge of the primary direction mode was chosen as TH, which causes early termination of the RDO process and the prediction of CU size.

The computation time complexity during RD cost estimation is increased since CTU partition requires an additional RDO technique. To tackle these negative aspects, (Karwowski & Domal 2016) proposed accurate probability estimation using Context Tree Weighting (CTW) techniques into CABAC algorithm with sophisticate data statistics modeling that exploits significantly higher number of binary trees Adaptive Context Tree Weighting (O'Neill et al. 2012) uses only the motion histogram information for optimizing the rate-distortion. So, on the whole, this affects the performance, and increases the complexity of the entropy encoder, while increasing memory requirement.

2.3 Inter Coding Techniques for HEVC

Lee et al. (2015) initiated a unit decision method based on block texture information to lower the computational difficulty of the HEVC range extension encoder. However, computation load on the encoder was still soaring.

In order to identify the district with the dominant gesture and saliency-based binary form for the current block, Podder et al. (2016) proposed an MD in the HEVC that matures a content-based versatile weighted expense capacity. Then again, due to the pre-processing stages of this technique, additional encoding time was needed that may reduce the computational time savings.

Li et al. (2017) presented an image feature-based strategy to efficiently decrease the CC of interprediction coding in HEVC. In this method, the general relocation of the largest coding unit (LCU) at the respective location between the nearby frames was determined using ME. The extent of the CU was determined by xCompress CU function of the encoder as that one didn’t require the approximation of RD expense meant for every level of the depth. However, the TH used in the inter prediction process for estimating the relative motion of the LCU was fixed.

2.4 Interprediction Mode Decision Techniques in HEVC

Purnachand et al. (2012) presented a fast ME algorithm for HEVC encoder, including pivoting hexagonal grids to find the global minimum. In this calculation, a versatile limit factor was used for an early termination. Conversely, the perseverance of the most optimum CU partition for individually CTU and the more superior PU mode for every CU root's higher computational intricacy.

Hsu & Hang (2014) presented a fast algorithm that comprises splitting decision (SD) and termination decision (TD) in creating the CU quadtree. Here, the CU-level swift decision was presented depends on the scrutiny in the temporal and SP neighborhoods. Hence, the candidate depth of CU was predicted based on the extent of its adjacent CUs and collocated CU(CCU). However, the efficiency of this algorithm was poor as PSNR was low, which degrades the video quality.

Jiang et al. (2018) designed an efficient CU size decision algorithm based on the probabilistic graphical models for HEVC intercoding. In this algorithm, two main methods were employed. CU size ET (CUET) decision approach and CU size early skip (CUES) decision approach, respectively. CU pruning was modeled as a binary classification issue, which was mainly based on the Naive Nayes (NB) model. Moreover, a Markov random fields (MRF) model-based method was presented to improve the algorithms performance by using the offline learning method which was used for obtaining statistical parameters.

However, it requires improvement on MRF model with neighboring CUs to further increase the accuracy of the CU size decision process. Known from the introduction, SSIM weighs in the structural distortions, such as noise, blocking artifacts, blurring and ringing, which are simply observed by the human eye. Although outcomes may vary between SSIM and PSNR, they often correlate in image compression conditions (Hore & Ziou 2010).

All the approaches mentioned above use either intra- or intercoding and some approaches did not satisfy the intra- or inter-PM decision. In the proposed frame work, considering both intra- and intercoding and PM decision, the proposed approach increases the video quality and reduces the CC.

2.5 Research Gap

Some of the limitations are identified from the previous researcher's related work is shown in Table 2.1. To overcome the issues identified in the conventional techniques and enhance the achievement of the HEVC encoder system, the proposed framework is required**.**

Table 2.1 Previous Related Works and Their Limitations

S. No.	Authors	Title Reference	Merits	Mode Decision Applied		Limitations
				Intramode	**Intermode**	**Demerits**
1.	Cho & Kim (2013)	[89]	Low computational complexity.	Yes	No	accelerate of the CU parting and pruning method cause coding effectiveness corruption because of expanded

						misclassification in the fragmented and non-fragmented CU choice of the early CUS and pruning tests.
2.	Jain & Rao (2014)	[90]	Reduced computational complexity i.e., encoding time.	Yes	No	It requires an adaptive TH to terminate the MD and further reduce the encoding time.
3.	Gan *et al.* (2015)	[91]	Less computational complexity and encoding time.	Yes	No	A considerable time reduction was achieved by reducing the PSNR and increasing the BR i.e., PSNR was not increased that degrades the video quality.
4.	Ding *et al.* (2016)	[92]	Higher-gain time saving and less computa	No	Yes	This algorithm should depend on a suitable motion search window.

			tional complexity.			
5.	Podder *et al.* (2016)	[93]	More appropriate for real-time video coding applications because of reduced overall average computational complexity.	No	Yes	Because of the pre-processing stages of this technique, an additional encoding time was needed that may reduce the computational time savings.
6.	Karwowski & Doma´nski (2016)	[94]	Increased compression gain and reduced BR.	No	No	The possibility estimation influences together the multifaceted nature and the memory request of a video decoder.
7.	Elyousfi (2014)	[95]	Reduced entire encoding time.	Yes	No	It is not appropriate for common Interprediction configurations.

8.	Xu *et al.* (2018)	[96]	Less computational complexity.	No	Yes	Low-motion regions for example, foundations still exist which influences the CUS decision.

2.6 Issues in Intra and Inter coding Prediction Mode in HEVC

The problems considered for this research work are as follows:

1. The utilization of Krichevsky–Trofimov (KT) estimators for CABAC-CTW causes high computational complexity and requires a number of memory buffer reservations.
2. The CABAC-CTW uses an angular PM for intra-prediction unit than directional modes that leads to high computational complexity.
3. Complete RDO methods are not utilized because of the complication and huge compression time.
4. The coding efficiency of conventional fast intra- and inter-prediction techniques require further improvement based on machine learning approaches.

2.7 Plan of the Research

The plan of the research work is given below:

1. The functional block diagram of HEVC is described and also compared with existing video codec.

2. The Efficient Image Coding Techniques based on seam Identification and Integer Wavelet Transform have been implemented.
3. Inter-prediction, and intra-prediction are achieved by proposing CNN based ensemble algorithms and LSTM.
4. The parameters of HEVC such as peak signal to noise ratio and structural similarity index are optimised.

2.8 Dataset

The proposed techniques were implemented and tested using MATLAB 2019b. The test conditions endorsed in the study are as follows!

- Software runs: Intel Core i7-2600 CPU @ 3.4GHz with 1TB memory.
- HM-14.0 for HEVC
- CTU: Size and extreme depth-: 8×8, 16×16, 32×32.
- In this reseach work, four QPs: 22, 27, 32 and 37 are available.
- Eight video sequences (VS) are considered which is publicly captured
- 30-60 Frames per Seconds used.

Table 2.2 Test Video Sequences

Class.	Size.	Sequence.
Class A.	(2460X1600).	PeopleStreet.
		Traffic.
Class B.	(1920X1080).	Kimono.
		ParkScene.
Class C.	(832X480).	BQMall.
		PartyScene.
Class D.	(416X240).	BasketballPass.
		RaceHorses.

The performance metrics used in this research are Peak Signal to Noise Ratio (PSNR), Structural Similarity Index (SSIM).

2.9 Summary

In this chapter, the related works for this research work were studied and the research gap between the proposed and existing techniques identified by many researchers using intra- and intercoding MD techniques for both intra- and interprediction techniques. From this survey, several issues were identified in conventional intra and intercoding prediction techniques for HEVC are discussed. Here, the primary goal of the research work was identified. To achieve the goal and also to overcome the challenges, the proposed plan for intra/intercoding and prediction in HEVC were made in this research work using CNN based Ensemble algorithm and Neural Network techniques.

CHAPTER-3

Efficient Image Coding Techniques based on seam Identification and Integer Wavelet Transform

In this chapter different types of image, image compression using seam carving and integer wavelet transform are discussed and Simulations were carried out based on the proposed method.

3.1 Image Definition and Types

To make an image, a two-dimensional function called F(x, y) is utilised, where x and y are spatial coordinates. The amplitude of F at any pair of coordinates represents the intensity of that image at that location (x, y). A digital picture is one in which F has finite x, y, and amplitude values. Following are different types of images.

Binary Image: A binary image consists of two pixel values: 0 and 1 which represent black and white respectively. Binary picture is also known as a monochrome image.

Black and White Image: A black and white image is one that has only black and white colours.

Colour Formats in 8 Bits:-Greyscale images are another name for this style of image. There are 266 distinct hues in total. In this style of image, 0 denotes black, 255 denotes white, and 127 denotes grey.

High Colour Image Formats: 16 bit colour formats are also known as High Colour Image Formats.

An analogue image is one that contains a continuous range of position and intensity data. An analogue image is made up of magnitudes that are constantly changing in space. CRT image is a good example. A digital image consists of picture elements called Pixels, whch are the smallest samples of an image.

3.2 Image Compression using Integer Wavelet Transform

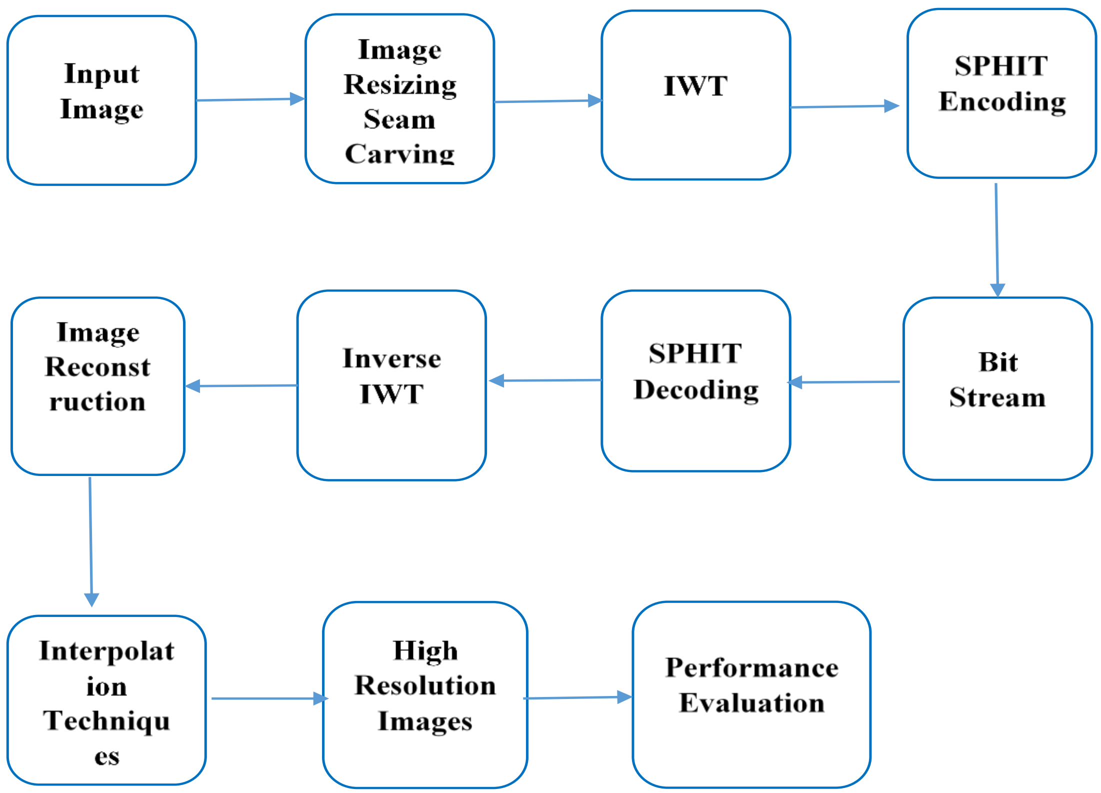


Figure 3.1: Block Diagram

Utilizing a wavelet transform, the compressed image is converted to the frequency domain. Images are split into odd and even components and then into four levels of frequency components in the wavelet transform. The image is then encoded using SPIHT coding, with the four frequency components being LL, LH, HL, and HH. The bit streams are then acquired. SPIHT decoding is used to decode the gathered bits. In order to create the compressed image, an inverse wavelet transform is eventually used.

3.3 Image Retargeting using seam carving

By removing a seam, or continuous path of pixels (vertically or horizontally), from an image, the user is able to resize the image. A horizontal seam is a line of pixels that constantly runs from the left to the right in an image, whereas a vertical seam is a path of pixels that runs continuously from the top to the bottom of an image.

Figure 3.2: Image with vertical seam

3.4 Implementation of algorithms

Calculating the gradient image for the original image is the first step in deciding whether to remove or insert a seam. The calculation of both horizontal and vertical seams frequently uses the gradient image. It can be created by either extracting the luminance channel from an HSV image or by averaging the gradient images for the R, G, and B channels. A gradient image is shown in Figure 3.3 as an example. Different gradient operators may be used, but the sobel operator was used to calculate the gradient image for this project.

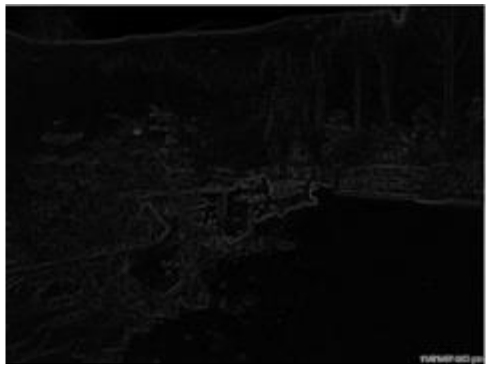


Figure 3.3: Gradient Image

The energy map image is calculated once the gradient image has been computed. In addition to needing to be recalculated after each seam removal, the energy map picture needs to be generated independently for either vertical (Figure 3.4) or horizontal (Figure 3.5) seams. For the vertical seam case, it is calculated as follows (a horizontal energy image can also be calculated using the same algorithm, but with a transposed input image): The value at (i,j) in the energy map is equal to the product of the current value at (i,j) from the gradient image and the minimum of the three nearby pixels in the previous row, i.e. min((i-1,j-1),(i-1,j),(i-1,j+1)) from the energy map, for each pixel (i,j) in the gradient image (see Table 1). When a pixel I j) is near the edge of an image, just (i-1,j) and either (i-1,j-1) or (i-1,j+1) are used, depending on whether (i,j) is on the right or left edges, respectively. The values in the energy map picture are set to those in the gradient image for i=1 (the first row).

Table 3.1: Pixel indices

(i-1,j-1)	(i-1,j)	(i-1,j+1)
(i,j-1)	(i,j)	(i,j+1)
(i+1,j-1)	(i+1,j)	(i+1,j+1)

Figure 3.4: Vertical Seam Energy map

Figure 3.5: Horizontal Seam Energy map

Following the creation of the energy map, the best seam can be identified by first locating the minimum value in the last row, which corresponds to the (i,j)th pixel, saving the location of the pixel for use in removal, and then working backwards to locate the minimum of the (i,j)th pixel's three closest neighbours in the (i-1)th row and saving that pixel to the seam path. The best seam is produced by continuing this process up until the first row; a sample of this seam is displayed in Figure 3.6.

Figure 3.6: Energy map with vertical seam

The seam line is deleted from both the gradient image and the original RGB image after choosing the best seam; the remaining pixels are then relocated to the right or upward to create a continuous image.

A smaller image with the same amount of scene content is left behind after repeating the process to remove a sequence of seams that run either horizontally or vertically. This is seen in Figure 3.7, which reduces an image from 640x480 pixels to 320x240 pixels. As can be seen, removing several seams causes artefacts in the final image.

Figure 3.7: Resized Image

The average of the two neighbouring pixels along the seam can be injected in the case of seam insertion, which enlarges the image. A seam can be calculated along a specified path. The requirement that the computation of the first N seams to be removed along a given direction needs to be completed before averaged pixels are inserted along each subsequent seam limits the maximum increase in image size in my implementation, which was previously discussed in the features and functionality section. In order to prevent repeatedly inserting pixels along the same seam, we utilise this algorithm to discover N seams. The two-dimensional DWT is a reasonably simple expansion of the one-dimensional DWT and is of particular importance for image processing and given in equation 3.1.

$$X_{a,b} = \int_{-\infty}^{\infty} x(t)\, \varphi_{a,b}(t) dt \tag{3.1}$$

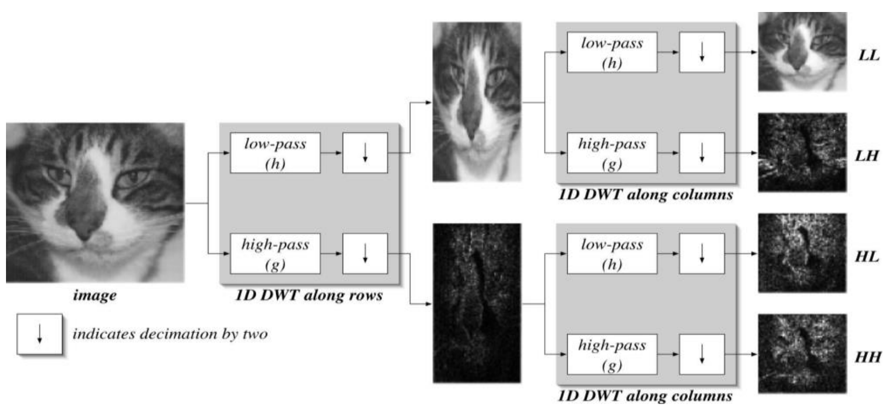


Figure.3.8: DWT

Two levels and two dimensions make up DWT. In the modified space, the single-dimensional DWT is applied to the rows and produces the columns, resulting in the four sub-band areas of LL, LH, HL, and HH.

Figure 3.8 depicts the fundamental one-level, two-dimensional DWT methodology. Starting with each image row, run a one-dimensional, one-level DWT is run. Then, we apply a one-level, one-dimensional DWT to the columns of the modified images from the first phase. Figures LL, LH, HL, and HH show separate images produced by these two techniques, each of which has four distinct bands. Low-pass and high-pass filtering are denoted by the letters L and H, respectively. In most cases, the LL band corresponds to a two-fold down-sampled version of the original image. While the HL band aims to preserve localised vertical elements, the LH band attempts to preserve localised horizontal characteristics of the source image.

Finally, the HH band isolates the image's limited high-frequency point characteristics. Unlike the one-dimensional DWT, two-dimensional DWT returns the picture's highest frequencies. More decomposition layers can be added to the converted picture's LL band to extract lower frequency characteristics.

3.5 Progressive Image Transmission

After the image pixels have been transformed into wavelet coefficients, SPIHT is applied. It is believed that the original image was composed of a set of pixel values, with the pixel locations (i, j). On the array, which is provided by, WT is applied.

$$c(i, j) = DWT\{p(i, j)\}. \quad (3.2)$$

The wavelet coefficients are represented by c (i, j).

The SPIHT decoder starts with a zero reconstruction vector and updates its components after receiving the coded message. The inverse wavelet transform, also referred to as "progressive transmission," can be used to reconstruct the image by the decoder once it has the value (approximate or precise) of some coefficients.

$$\hat{p}(i, j) = IDWT\{c(i, j)\} \qquad (3.3)$$

One of the main goals of a progressive transmission system is to send the most important information first, which has the biggest impact on reducing distortion. The mean squared error (MSE) distortion measure is used to make this decision.

$$D_{MSE}(p - \hat{p}) = \frac{1}{N}\left\| p - \hat{p} \right\|^2 = \frac{1}{N}\sum_{i}\sum_{j} (p_{i,j} - \hat{p}_{i,j})^2 \qquad (3.4)$$

Where $p_{i,j}$ is the original pixel value. N is the number of pixels in an image and $\hat{p}_{i,j}$ is the pixel value that has been rebuilt.

3.6 Inverse Discrete Wavelet Transform

The reconstruction of IDWT, like DWT, may be explained using filter bank theory. It's a matter of reversing the process. The DWT coefficients are up sampled first by inserting zeros between each coefficient, thereby doubling their lengths. The detail coefficients are then convolved with the reconstruction wavelet filter, while the approximation coefficients are convolved with the reconstruction scaling filter. The data are then combined to obtain the original signal.

Discrete Wavelet Transform of approximation coefficient (CA) is :

$$\mathrm{W}_{\varphi}(\mathrm{K},\mathrm{S}) = \frac{1}{\sqrt{\mathrm{N}}}\sum_{\mathrm{t}} \mathrm{f}(\mathrm{t})\, \emptyset_{\mathrm{K,N}}(\mathrm{t}),\ \ \emptyset_{\mathrm{K,S}}(\mathrm{t})\text{: Scaling finction} \tag{3.5}$$

and detail coefficients are

$$\mathrm{W}_{\varphi}(\mathrm{K},\mathrm{S}) = \frac{1}{\sqrt{\mathrm{N}}}\sum_{\mathrm{t}} \mathrm{f}(\mathrm{t})\, \varphi_{\mathrm{K,s}}(\mathrm{t}),\ \ \varphi_{\mathrm{K,S}}(\mathrm{t})\text{: Wavelet function} \tag{3.6}$$

IDWT

$$\mathrm{F(t)} = \frac{1}{\sqrt{\mathrm{N}}}\sum_{\mathrm{t}} W_{\emptyset}(K,S)\, \emptyset_{\mathrm{K,S}}(\mathrm{t}) + \frac{1}{\sqrt{\mathrm{N}}}\sum_{\mathrm{t}} W_{\varphi}(K,S)\, \varphi_{\mathrm{K,s}}(\mathrm{t}) \tag{3.7}$$

Before convolving to get the original signal, we must make our dwt coefficients periodic, just as we did with the signal before completing our DWT calculations on it. The first N/2-1 coefficients from the DWT coefficients are attached to the end to achieve this. Here scaling filter has a length of N. To get the segment of the signal, after convolution and addition, we collect the coefficients from N to the length of the signal + N -1. As a result, we get back to the previous signal.

3.7 Simulations and Results

Measures of Image Quality

Peak signal to noise ratio (PSNR) and mean square error are used to evaluate the reconstructed image's quality (MSE). The variance of q2 reconstruction error is also known as

MSE. The decoder calculates the MSE as follows between the original picture f and the rebuilt image g:

$$\text{MSE} = \sigma_q^2 = \frac{1}{N}\sum_{j,k}(f[j,k]-g[j,k])^2 \qquad (3.8)$$

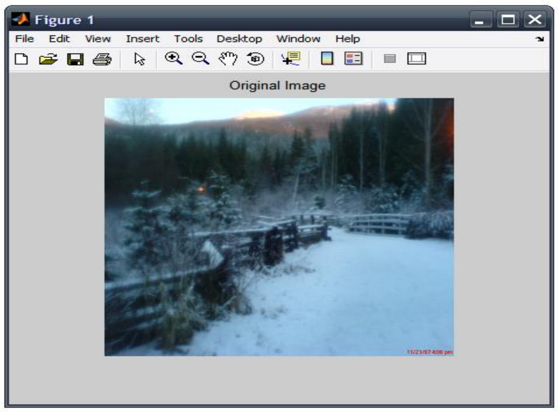


Figure 3.9: Original Image

N stands for the total number of pixels in each image, whereas j and k stand for the total number of pixels in the image. The ratio of signal variation to reconstruction error variance is known as PSNR. Following is a formula for calculating the peak signal to noise ratio between two photographs with an 8-bit per pixel resolution. Decibels are used as the measurement unit.

$$\text{PSNR} = 10\log_{10}\left(\frac{255^2}{MSE}\right) \qquad (3.9)$$

When PSNR reaches 40 dB or greater, the original and reconstructed images are essentially indistinguishable to the human eye.

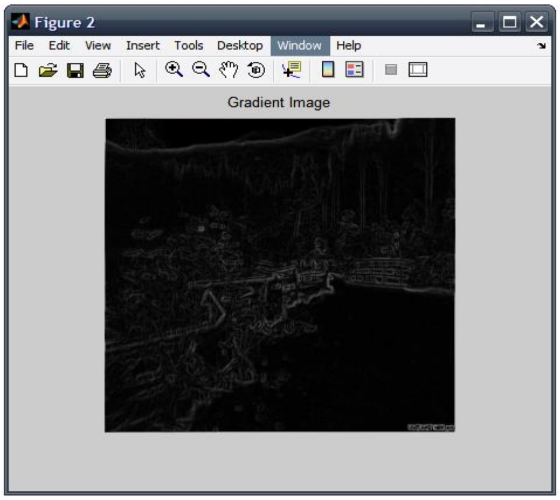


Figure 3.10: Gradient Image

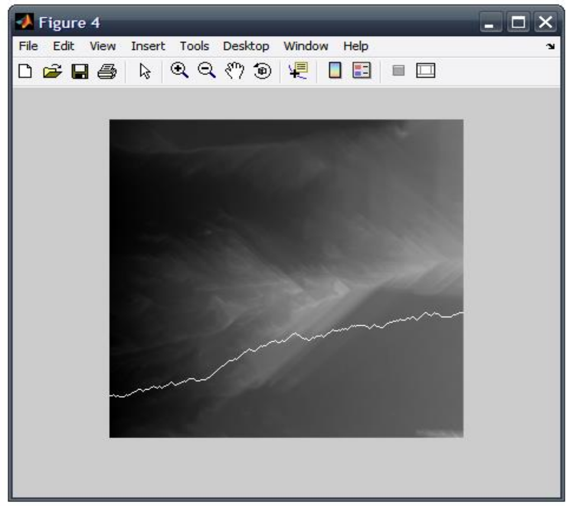


Figure 3.11: Energy map

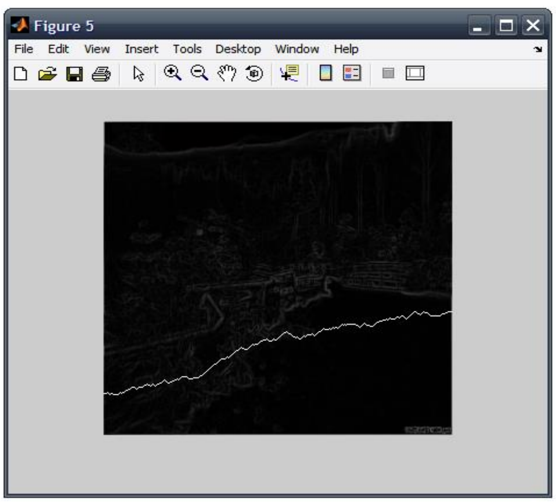


Figure 3.12: Horizontal Seam Carving

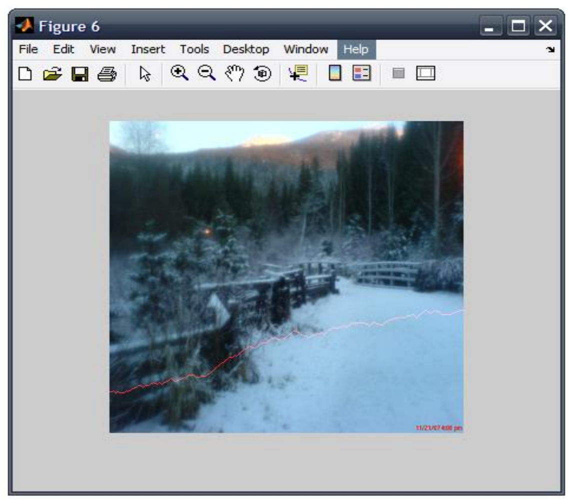
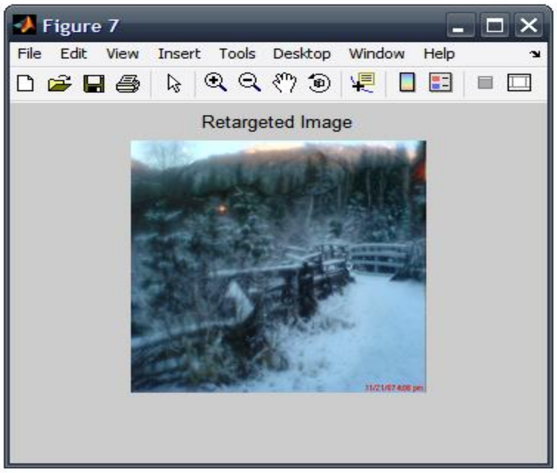


Figure 3.13: Seam carved retrieved image



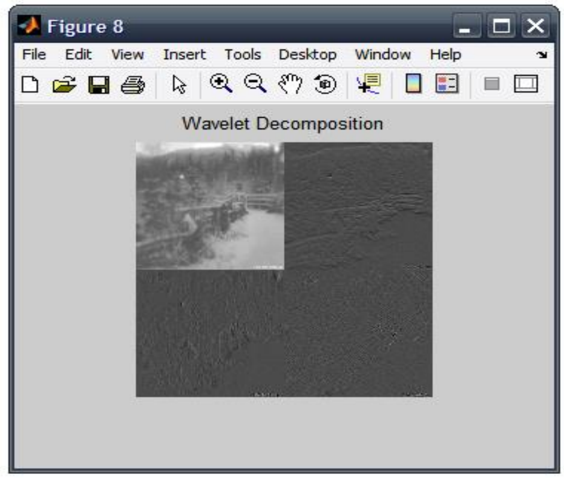


Figure 3.14: Retargeted Image

Figure 3.15: Wavelet decomposition

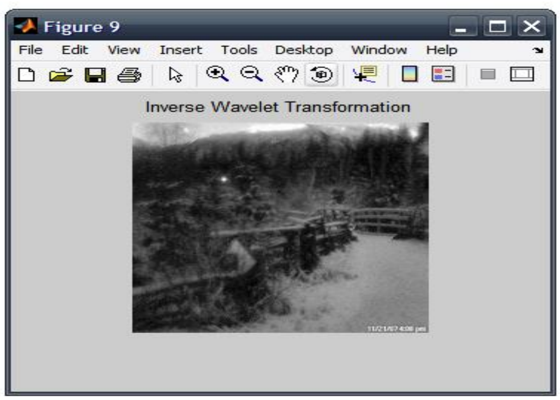


Figure 3.16: Inverse wavelet transform

Pixels positions are encoded for a horizontal or vertical seam either sarting from left to right or top to bottom, and to identify the places, just coordinates are necessary.

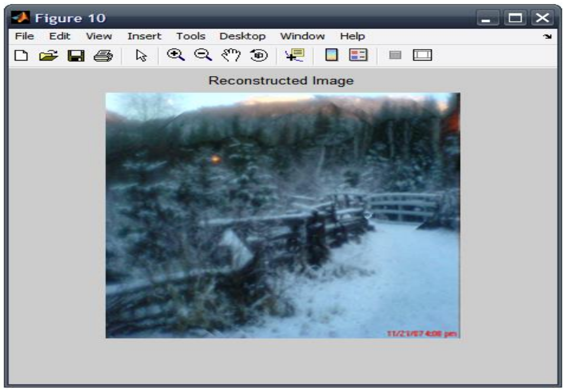


Figure 3.17: IWT Reconstructed Image

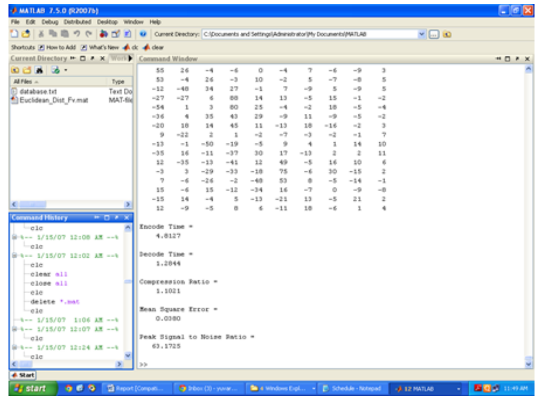


Figure 3.18: Results

The initial broadcast seam's state must be represented by one extra bit due to the modified seam transmission sequence. The seam block unit in each N X M input picture with L-scale DWT is 2l X 2l in size. In this case, N/2l and M/2l positions are encoded in the first pair of vertical and horizontal seams, and for each further pair of seams, the number of positions to be encoded is reduced by one. Figure 3.9 to Figure 3.18 illustrate image compression using seam carving and integer wavelet transform. Seam carving and integer wavelet transform based image compression technique provided mean square error about 0.036 and peak signal to noise ratio of 63.375 db.

3.8 Summary

In this chapter, different types of image and image compression techniques were discussed. This chapter also explains seam carving and integer wavelet based image compression techniques and its results.

CHAPTER-4

High Efficiency Video Coding Architecture

In this chapter, basic terminologies used in high efficiency video codec such as prediction unit, coding units and different coding tree units block size are described. The chapter also discusses performance comparison of high efficiency video codec with different video codec unit.

4.1 Introduction

Video compression and decompression or codec algorithms have been around for four decades. They have become a necessity in today's era, owing to the ever- increasing resolution capabilities of video cameras and their increasingly high storage and transmission requirements. It can be seen that while an average 1080p video of length 1 minute would turn out to be approximately 130MB in size, the needed bandwidth to stream such videos is only between 8 to 16 Mbps. This can be attributed to advancements in video codec algorithms. The High-Efficiency Video Coding standard/ H.265 is one of the latest standards in video encoding designed by MPEG as a successor to the H.264/ AVC standard. It promises a 25 percent to 50 percent reduction in bit rate without compromise in video quality. Though similar in architecture, the improved features in the H.265 include the use of CTUs instead of macroblocks, better in-loop filters, etc., leading to greater accuracy in the encoding process with a reduction in bit rate

The general representation of the image in a computer is like a vector of pixels. A pixel is an abbreviation of picture element. Thus we can infer that a picture is generated by picture elements or pixels just like a molecule is made up of atoms. When we talk about resolution of a picture, we are actually talking about the pixels it has. For ex: A 200 X 200 image can be seen as a square of side 200. Each pixel has representation in terms of bits, suppose if there are 4 bits for each pixel, the size of a 200 X 200 image with 4 bit pixel would be 200*200*4 => 160000 bits or approximately 20kB.

This was for an image and HEVC is a coding standard for videos; so what is a video? We can say that video is a sequence of images and hence each frame of a video is an image and therefore if can compress those images, we can eventually compress the video. The frame rate is the number of frames passing per second in a video and so more the frame rate, smoother the video. Video compression involves spatial image compensation and temporal motion compensation. Video Compression is needed mainly due to low channel capacity hence transferring along channels becomes too inefficient and thus hence compression saves both time and space at the cost of computational complexity. Thus it fastens the file transfer process and also reduces space on disk.

If we look at internet traffic, it is mainly due to video streaming. Approximately 80% of internet traffic is due to videos and 20% is due to other data. In such a scenario, where videos dominate the traffic over internet, we need to find more efficient video coding standard which is faster, saves bandwidth and is cost and quality efficient.

Figure 4.1: Video and data Traffic over Internet

4.2 High Efficiency Video Coding Terminology

The basics of H.264 are worth noting before we move on to the features of HEVC, as HEVC (standardized in 2013) is ultimately a better version of H.264.In the past we had the MPEG standard in DVDs in 1996 for video coding and now we have moved to H.265 with technological advancements. Every standard promises to fulfill the same or even better quality with lower cost and bandwidth. There are two main methods in video compression, one is the Inter-frame and another is Intra-frame.

In the Interframe method, we basically compare the previous and future frames with the current one and we only encode what is changed, for example, in a 5sec movie clip while the actor is reciting a poem, only his facial expressions or his gestures change while the background is the same; here we only have to encode the actor and not the background, so the background data could be saved in many frames.

On the other hand, in Intraframe, we look for similarity in the adjacent pixels within a frame.We initialize with an I-frame which is likely to be stored as a JPEG and then we divide it into small 16 X 16 pixels which were called Macro blocks previously. Then we move on to the next frame and compare its macro block with the I frame; if some blocks are roughly same, then we give this block the status of Predicted frame or P-frame (this was interframe). Next we give the completely new pixel values for pixels which have changed only for this frame, and this way we intracode the intercoded block.

In the recent H.265, instead of a 16 X 16 macroblock, we have a 64 X 64 Coding Tree Unit. Now this is a significant difference because in H.264 standard in 2003, 1080p was the greatest milestone achieved but now we have achieved 4k and therefore more efficient video coding is possible and this is made possible with larger coding tree units. Now the second improvement is in the Intraframe compression. In HEVC, we have more prediction direction compared with previous standards and this enables more precise compression with more options in various directions for pixel values.

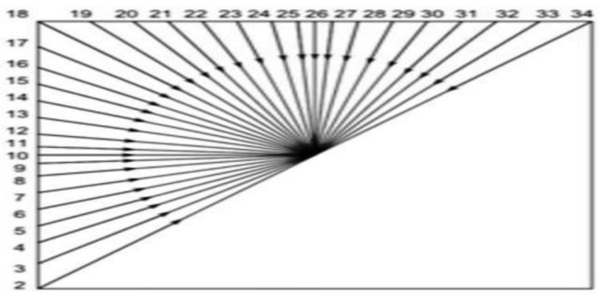


Figure 4.2: Angle definitions of angular intra prediction in HEVC for 2 to 34 modes and the associated displacement parameter H.265/HEVC Video Coding

Actually we can further breakdown our CTUs to Coding Units or Coding Blocks(CBs) which can have 8 X 8 pixels and these CUs can be further partitioned in different. PredictionBlocks.(PBs).

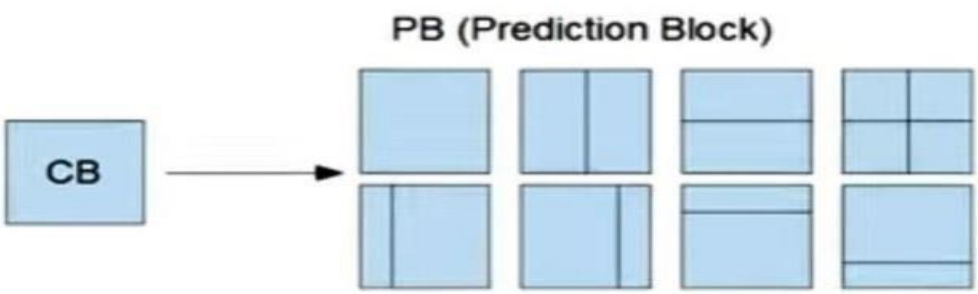


Figure 4.3: Breaking of CTU into CBs and PUs

The motive behind the prediction block is to mathematically generate pixel values instead of storing them and this gives an edge for compression by reducing the size of each frame consecutively. For instance, look at the PU of 4 X 4 surrounded by two groups of pixels, A and B We can use various Intraframe prediction models which the HEVC standard offers.

Figure 4.4: PU of 4x4 surrounded by two group of pixels

There are different prediction models for different categories, for example, if we just want the background of a single color, we can opt out DC type or if we want a pattern to be continued, then we can choose Angular. Previously, there were only nine prediction modes and thus quality deterioration was more critical but now in H.265/HEVC, we have35 prediction modes which is more than 3 times what was available in H.264. This therefore gives highly intra-compressed frames with better quality.

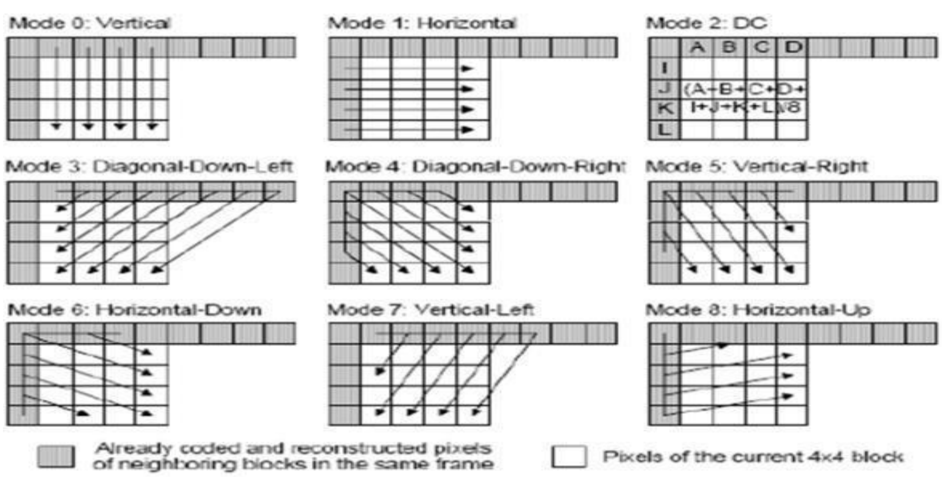


Figure 4.5: Prediction of modes of HEVC

The main points which demarcate HEVC from H.264 are:

The macroblocks are renamed Coding Tree Units (similar in structure), the previous macroblocks had a size range of 16 X 16 but the new CTUs have a size range of 64 X 64. The larger range of size for HEVC provides more clarity and smoothness. There are 35 prediction modes in H.265, as compared with only 9 modes in H.264.

4.3 Block Diagram of High Efficiency Video Coding

Starting from the very basic process of a video codec, we have the following flow diagram:

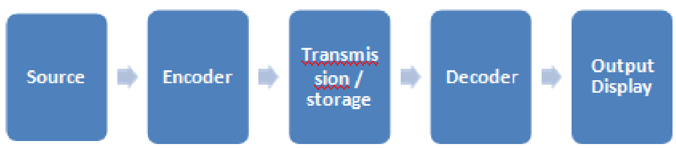


Figure 4.6: Video CODEC

It is worth noting that Encoder and Decoder together constitute a “Codec”. Each process in encoding has a counterpart in decoding. The previous H.264/AVC gave ~2X better compression than MPEG and the recent H.265/HEVC is also ~2X better than H.264; thus, with each passing standard, the compression rate is increasing while the size is reducing, with a better picture quality. Size (H.265) ~ 0.5 Size (H.264).There is a kind of trade-off in video coding:

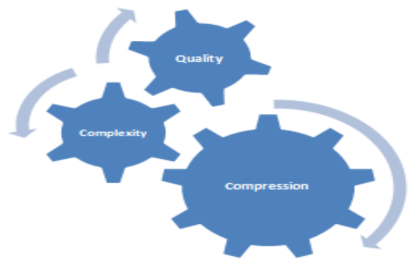


Figure 4.7: Video Coding trade-offs

So, we can achieve a better compression rate with better picture quality by increasing computational complexity. Starting with detailed algorithm of H.265 video code, there are some terms which we have already seen in the introduction; therefore the algorithm can be interpreted more effectively. Before moving on to the complex algorithm one needs certain insights to understand the principle behind the algorithm.

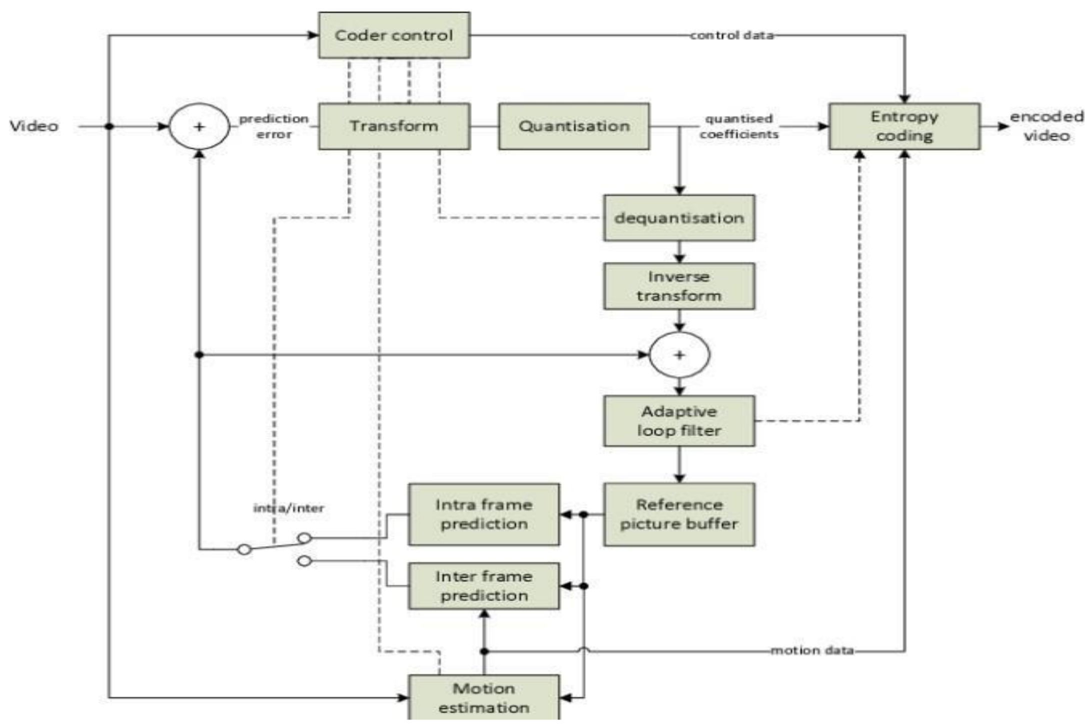


Figure 4.8: Block Diagram of H.265 Video

Partitioning

It refers to breaking up the video frame into small units. A frame or picture can be broken down into slices and each slice is made of several CTUs. These CTUs can be further broken down into coding units. A video codec processes one CTU at a time. For ex: Take a frame of a video and now take a macro block or CTU from it, each CTU will have a luma (brightness) component and red and blue color differentiating components, which are CUs. The brightness component is stored at a higher resolution and the color components are stored at a lower resolution as the human eye is more sensitive towards brightness than to colors.

Prediction

The prediction is made with respect to the current frame and very little residual is left on comparing the original and the predicted frame. There are two kinds of prediction, one is Interframe and the other is Intraframe compression. We cut down the undesired information stored by mathematically preceding it based on various prediction modes Inframe: N x N, 2N x 2N

Inframe: N x N, 2N x 2N, dN x N, N x dN There are in total eight methods for partitioning an interceded coding unit and two main methods are merge and advanced motion vector prediction. On the other hand there are 35 modes of interceded in which 33 are uniquely patterned where one is planar and one is DC, which predicts PB by filling in the average of surrounding pixel values.

Transform + Quantize

Base on Fourier transform, Laplace transform and Z transform, the basic idea of transforming a compressed video after prediction is also the same. The images or frame’s blocks are converted to frequency domain representation followed by quantization where removal of unnecessary small values takes place. These two steps reduce the size considerably by converting the image into frequency domain and further quantizing it to discrete integers.

Entropy Encoding

This concept is derived from the concept of Information Theory used in Digital Communication Systems. Entropy Encoding is done in order to convert the quantized block values to binary form. This is done by assigning binary values to a series of information and thus each specification takes its corresponding binary value. This is also referred as CABAC which stands for Context Adaptive Binary Arithmetic Coding which is a sophisticating and complex content management scheme. After this step the memory requirement is reduced and therefore better compression is achieved leading to increase in speed of transmission. If all these steps are done precisely, then after decoding you can get a video very much similar to the source. A decoder simply does the similar steps in reverse manner to give the output which is very similar to the source input but highly compressed in nature. A video file of size in GBs can be easily compressed to a file of size in KBs where even the changes in the compressed version are non-noticeable.

Adaptive Filters

An Adaptive filter is essentially a digital non linear filter with self-Adjusting characteristics. It adapts, automatically, to changes in its input signals. Contamination of a signal of interest by other unwanted, often larger signals or noise is a problem encountered in many applications. Where the signal and noise occupy fixed and separate frequency bands, conventional linear FIR filters with fixed coefficients can be used to extract the signal. But when there is a spectral overlap between the signal and noise and the band occupied by the noise is unknown or varies with time, fixed coefficient filters are inappropriate [118].

Motion Estimation

In HEVC Successive video frames may contain the same objects (still or moving). Motion estimation examines the movement of objects in an image sequence to try to obtain vectors representing the estimated motion. Motion compensation uses the knowledge of object motion so obtained to achieve data compression. In interframe coding, motion estimation and compensation have become powerful techniques to eliminate the temporal redundancy due to high correlation between consecutive frames [118].

In real video scenes, motion can be a complex combination of translation and rotation. Such motion is difficult to estimate and may require large amounts of processing. However, translational motion is easily estimated and has been used successfully for motion compensated coding.

Most of the motion estimation algorithms make the following assumptions:

1. Objects move in translation in a plane that is parallel to the camera plane, i.e., the effects of camera zoom, and object rotations are not considered.
2. Illumination is spatially and temporally uniform.
3. Occlusion of one object by another, and uncovered background are neglected**.**

Summarizing all this, we have portioned the frames into various blocks, then we predicted the blocks after which we transformed the image values to frequency domain representation followed by quantization, and at the end entropy encoded for good transmission speed with much less memory requirement. Furthermore, we can see a more detailed H.265 coding standard where we have separately shown all the inter frame predicted filters.

4.4 Performance Comparison of high efficiency video coding

Table 4.1: H.264 vs. H.265

Parameter	**H.264/AVC**	**H.265/HEVC**
Names	Advanced Video Coding MPEG-4 part 10	High Efficiency Video Coding MPEG-H part 2
Approved Date	2003	2013
Progression	Successor to MPEG-2 part as known as H.222/H.262	Successor to H.264/AVC or MPEG Part-10
Key Improvement	1.The decline in bit rate compared with MPEG-2 Part is 40% -50% 2. Available to deliver High Definition Sources for online and Transmit	1.40%-50% Decline in bit rate at the same visual quality compared with H.264 2.It can used to implement Ultra High Definition.2K,4K FOR Online and Broadcast
Support Up to 8K	No, It supports up to 4k only	Yes
Support up to 300fps	No, It supports up to 59.94 Kfps.	Yes

4.5 Application of high efficiency video coding

1. Camcoder
2. Broadcast
3. Content Production and Distribution
4. Digital Camera
5. Internet Streaming, Download and Play
6. Medical Imaging

7. Mobile Streaming

8. Storage Media

9. Wireless Display

10. Remote Video Surveilance

11. Video Conferencing

12. Videophone

13. Telepresence

14. Digital Cinema

15. Home Cinema

4.6 Summary

This chapter discuss video codec architecture and different terminologies used in video codec. This chapter also discuss High Efficiency Video Codec Architecture, its performance compared to existing video codec and its applications which will be used for further enhamcement.

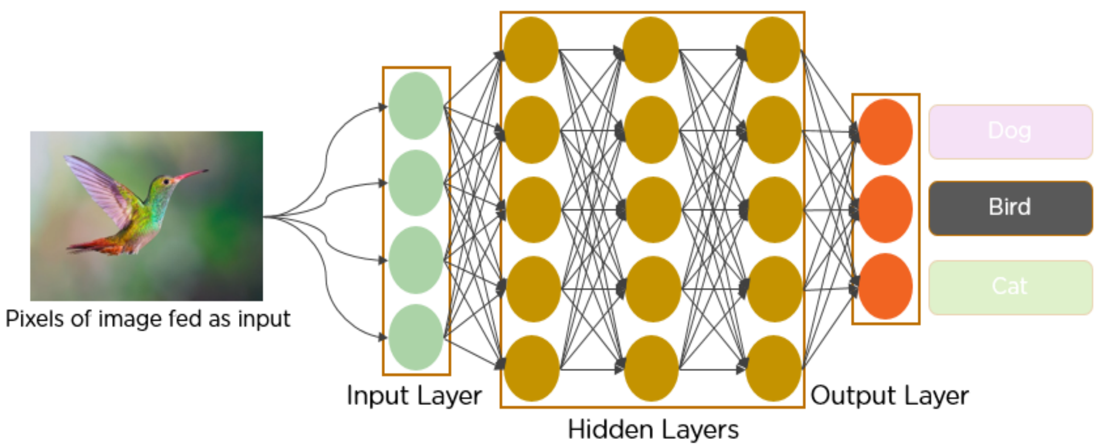
CHAPTER-5

Intra-Frame Prediction Using CNN based Ensemble Algorithms

This chapter presents detailed information about convolutional neural network and different layers of convolutional neural network. It also discusses the proposed convolutional neural network based ensemble algorithms and simulation results for inter frame prediction of high efficiency video code.

5.1 Introduction

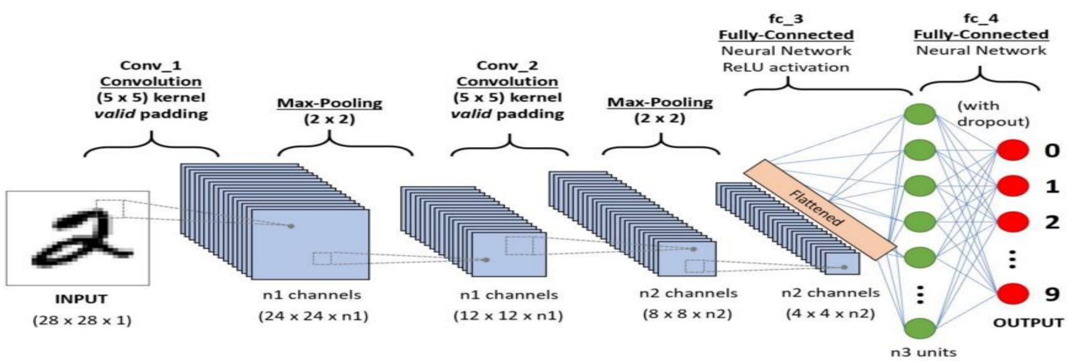
The use of deep learning to process massive amounts of data is the need of the hour. The popularity of hidden layers has overtaken that of conventional methods in the field of pattern recognition. A well-known variety of deep neural networks is convolutional neural networks.



Researchers have been attempting to create a system that can comprehend visual input since the 1950s, when AI was still in its infancy. In the years that followed, this discipline was referred to as computer vision. Computer vision advanced dramatically after a team of academics from the University of Toronto created an AI model in 2012 that outperformed the best photo recognition algorithms by a significant margin. CNN is a type of neural network that mimics human vision. Throughout history, CNNs have proved to be a vital component of many Computer Vision applications.

History of Convolutional Neural Network

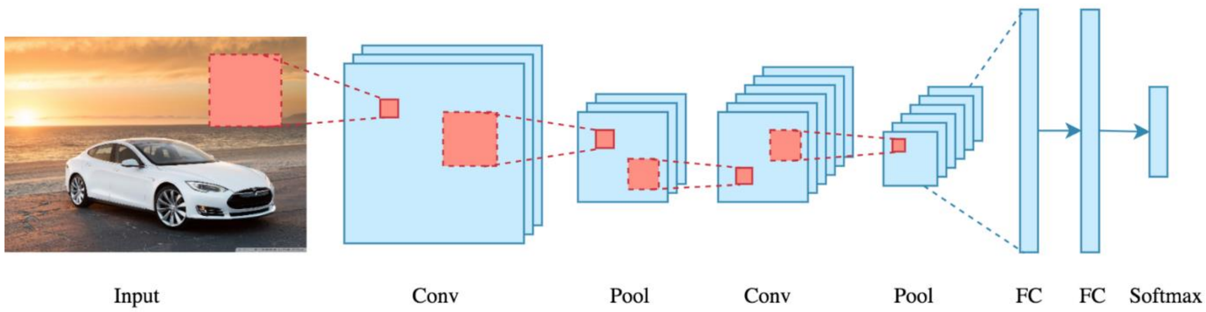
Convnet was first utilised in the United States in the 1980s. It was employed to identify written digits at the time. It was mostly used in the postal business to read zip codes, pin numbers, and other codes of this nature. The most important thing to remember about CNNs is that they require a lot of data and processing power to train. It was CNNs' primary drawback at the time, and as a result, CNNs were best restricted to the postal sector.



Alex Krizhevsky decided in 2012 that the department of deep learning, which uses multi-layered neural networks, needed to be revived. Researchers were able to rebuild CNNs because to the availability of large collections of data, including Image Net datasets containing hundreds of thousands of annotated photographs and an abundance of computational resources.

5.2. Convolutional Neural Network

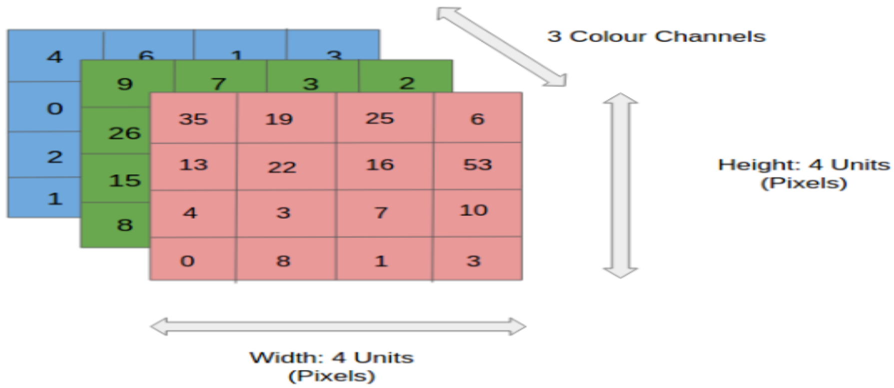
CNN is a class of DNN, most usually implemented to investigate visible images.We assume matrix multiplications while considering a neural network. In the case of Convent, however, a specific approach called Convolution is used. Convolution is a mathematical method that combines two functions.



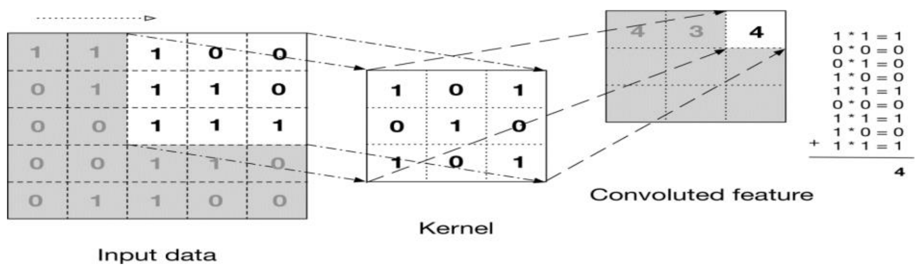
Finally, the Convent must compress the image into a format that is easier to analyse while preserving crucial information for forecasting.

Working of CNN

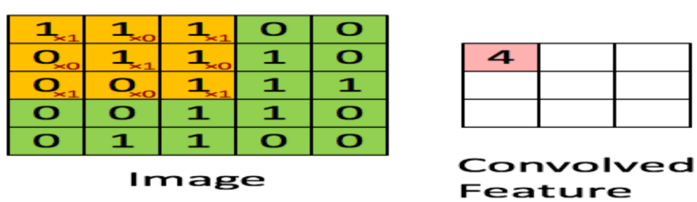
One needs to consider the basics of a picture and how it's represented before learning how CNN works. An RGB picture is a three-plane matrix of pixel values, while a grayscale image is a single-plane matrix.Consider following Figure.



Let's have a look at some grayscale photographs to see CNN functions.



In the figure above, one can see a convolution. We apply a filter/kernel to the input picture to get the convolved feature. The mixed feature of the next level is the same as the previous one.

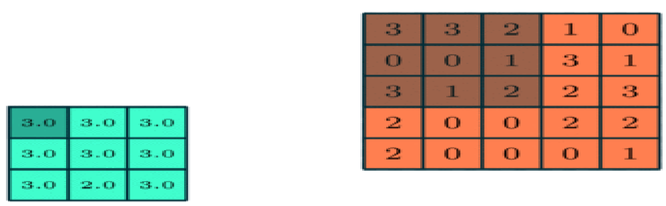


Multiple layers of synthetic neurons make up convolutional neural networks. Similar to real neurons, artificial neurons compute the weighted sum of a number of inputs and output an activity value. Each layer has a set of activation routines that are transmitted down from layer to layer when you load an image into a convent.

Typically, the first layer recovers fundamental details such as edges that are horizontal or diagonal. The following layer receives information and is tasked with identifying more intricate characteristics like corners and combinational edges. As we go deeper into the network, it becomes more capable of recognising complicated objects, faces, and other aspects.

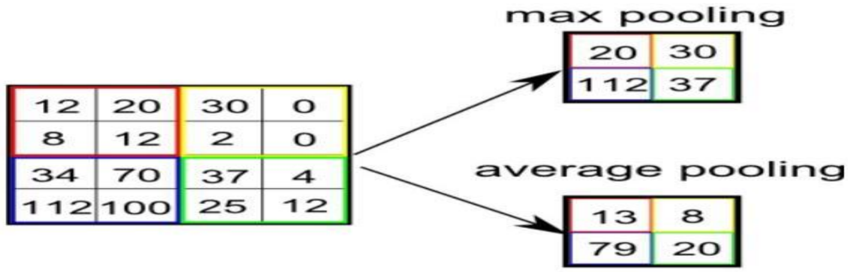
5.3 Pooling Layer

The Pooling layer, like the Convolutional Layer, is in charge of shrinking the Convolved Feature's spatial size. By limiting the data, the amount of computing power needed to process it is decreased. Average pooling and maximum pooling are two types of pooling.



Max Pooling determines a pixel's maximum value from a kernel-covered region of the picture. Max Pooling also acts as a Noise Suppressant. It de-noises and reduces the dimensionality of the data by deleting all noisy activations.

Average Pooling returns the average of all values in the image's Kernel region. As a noise reduction approach, average pooling considerably reduces dimensionality. We may therefore confidently state that Max Pooling performs better than Average Pooling.



5.4 Pros and Cons of Convolutional Neural Network

Pros

1. Learning accurate pattern and insights from the provided data.
2. For better and accurate results one can change network.
3. If properly configured and fed a large quantity of data, it can outperform other machine learning algorithms.

Cons

1. It requires more computational power
2. It consists of complex architecture most of the time.

5.5 Applications of Convolutional Neural Network

1. Image recognition.
2. Video Analysis.
3. Natural Language Processing.
4. Anomaly Detection.
5. Drug Discovery.

6. Checker Games

5.6 Ensemble Algorithms

Ensemble Algorithms are a form of machine learning algorithms that combine several different base models into a single best-fit predictive model. Ensemble Methods are Sequential and parallel ensemble techniques are the two most used forms of ensemble methods. Base learners are developed sequentially using adaptive boosting and other sequential ensemble techniques. The production of fundamental learners one after another promotes fundamental learners to rely on one another. Then, by giving previously misrepresented learners more weight, the model's performance is enhanced.

Fundamental learners are built in parallel in parallel ensemble systems like random forest. They promote the independence of basis learners by using concurrent production of basis learners. The independence of base learners considerably reduces the error caused by the use of averages. Since most ensemble techniques in base learning employ a single algorithm, all base learners are homogeneous. Those with identical traits and those belong to that the same type are homogenous base learners. Other approaches employ heterogeneous base learners, producing heterogeneous ensembles. Heterogeneous base learners are made up of various types of learners.

The practise of aggregating data using bootstraps is referred to as "bagging." Classification and regression are two of its most common uses. It employs decision trees, which dramatically minimise variance, to improve model accuracy. Reduced variance increases accuracy by preventing overfitting, a major problem with predictive models.

There are two methods for bagging: bootstrapping and aggregation. Bootstrapping is a method of sampling that use a replacement strategy to collect samples from the whole population (set). The sample with replacement approach increases the unpredictability of the selection process. To complete the technique, the sample is fed into the basic learning process.

Aggregation is used in bagging to account for all potential prediction results and to randomise the outcome. Because all outcomes will not be taken into account without aggregation, projections will be incorrect. As a result, probability bootstrapping methods or the sum of all prediction model outputs are used to aggregate the data.

Bagging offers the benefit of integrating weak base learners into a single strong student who is more stable than individual learners. It also eliminates any variance, resulting in lower model overfitting. One of its disadvantages is the computational expense of bagging. When the proper bagging process is missed, models may become more biased.

Boosting is an ensemble strategy that learns from prior predictor failures to improve future predictions. A number of weak base learners are combined into a single strong learner, resulting in a more predictable model. Boosting is the technique of putting together a group of weak learners in a certain sequence so that they can learn from one another and produce more accurate predictions.

Boosting strategies include gradient boosting, Adaptive Boosting (AdaBoost), and XG Boost. AdaBoost uses decision trees to train weak learners, which generally have one split, nicknamed decision stumps. In Ada Boost, the primary decision block is made up of similar-weighted observations.

Gradient boosting adds predictors to the ensemble in a sequential order, with preceding predictors correcting their successors and improving the model's accuracy. New predictors are fitted to counteract the consequences of prior forecasters' mistakes.The gradient booster employs the gradient of descent to aid in the detection and correction of errors in learners' predictions.

XG Boost uses decision trees with higher gradients to increase speed and performance. It heavily depends on the performance and speed of the target model's computations. Gradient boosted machines take a while to set up because model training must be done in a precise order.

Another ensemble strategy is stacking, also referred to as stacked generalisation. This method functions by enabling a training algorithm to combine the predictions of numerous learning algorithms that have a common characteristic. Regression, density estimates, distance learning, and classifications have all benefited from stacking. It can also be used to determine how frequent bagging errors are.

5.7 CNN Based Ensemble Algorithm

The learning problem is envisioned as a better model of the relationship between a block, maybe preserving a texture, and its local values that satisfy its expectations with the assistance of neural networks. It's worth mentioning that in [62], neural networks were successfully tested for intrablock prediction. In this situation, [62] employed completely connected neural networks to analyse blocks of sizes 4x4, 8x8, 16x16, and 32x32 pixels. In this scenario, consider both fully related and CNN blocks.

While totally associated neural networks perform well for small block sizes, CNNs are better for large block sizes, both in terms of prediction and PSNR-rate execution benefits. The neural network's choice is block size dependent, hence it shouldn't be sent to the decoder. This CNN configuration has been implemented in an H.265 codec/Matlab.This study’s contributions are as follows:

a) For intra-image prediction, proposed a completely associated CNN.

b) Demonstrate that, because of higher block sizes, CNN produces more precise and entirely associated predictions.

c) Convolutional neural networks can adapt well to changing circumstances when trained with masks of various sizes. In H.265, the position of the considered prediction unit inside the coding unit and within the coding tree unit determines the accessible context and hence the amount of known pixels in the region.

d) Demonstrate a preliminary intra prediction neural network ensemble strategy: they should not be trained in distorted contexts, as neural networks trained on undistorted contexts function well in distorted circumstances.

5.7.1. Proposed CNN-Based HEVC Intra Frame Coding Framework

The proposed CNN model is employed at the CTU level of the HEVC intra frame encoder, which is 64X64 pixels in size. Each CTU is first encoded using intra prediction in HEVC intra frame coding. After residual coding, the bit stream of the CTU is produced using entropy coding, and the CTU is rebuilt as a reference block for internal prediction of the next CTU to be encoded. The proposed CNN upgrade mode is activated, allowing the learned CNN model to forecast the rebuilt CTU's residual and improving the reconstructed CTU's nature. As illustrated in Figure 5.1, the projected residual and the original CTU are combined to create a new reconstructed CTU.

5.7.2 Classifier Mode Choice

The sum of square error (SSE) is used in the rate-distribution optimization (RDO) during the mode choosing process as an objective estimation to provide a better reconstructed CTU. A second signal, cnn flag, is included in the output bit stream that signals whether the CNN augmentation mode should be used. The suggested CNN based on sign would improve the character of each rebuilt CTU on the decoder side. As a result, by focusing on intra prediction accuracy, the improved replicated CTU improves coding efficiency in the proposed CNN learning-based structure, while the CTU serves as a reference for various squares.

A module containing a CNN model is executed and used in the HEVC encoder programme before intra prediction. The best CTU division results come from CNN classifier. The recommended work encoding strategy does not require numerous rounds to establish the optimal CU depth when using CNN encoding. The CU grouping calculation will help with intra encoding computational decrease. This means that the RDO process' hardware area can be created in intra coding mode.

5.7.3. Ensemble Learning through CNN

The intra predictions ensemble learning component of the framework leverages information to obtain reasonable predictions before applying the final deconvolution to rebuild HR images. Using a deep CNN, this is accomplished with ease. This module's contribution is a multi-channel image with several intra predictions, each of which may be thought of as a single channel for the related image. It's challenging enough to reduce this multi-channel image to a single channel by making plausible internal forecasts.

Consider this interaction as labelling, in which each block patch is selected from a collection of discrete Markov random field improvement procedures (MRFs). Correct expected functions, on the other hand, must be defined. The process of establishing final intra predictions from a huge number of learning candidates may not be easy, and algorithm-based movies may not perform well like typical regular films. It's possible that actions in close proximity are essentially nonlinear.

Furthermore, constructing and optimising functions do not ensure that their HR outputs may be successfully deconvolved for a reconstruction with better aesthetics. If the image doesn't meet the requirements of the convolution model, visual anomalies like ringing appear in the replicated HR image.

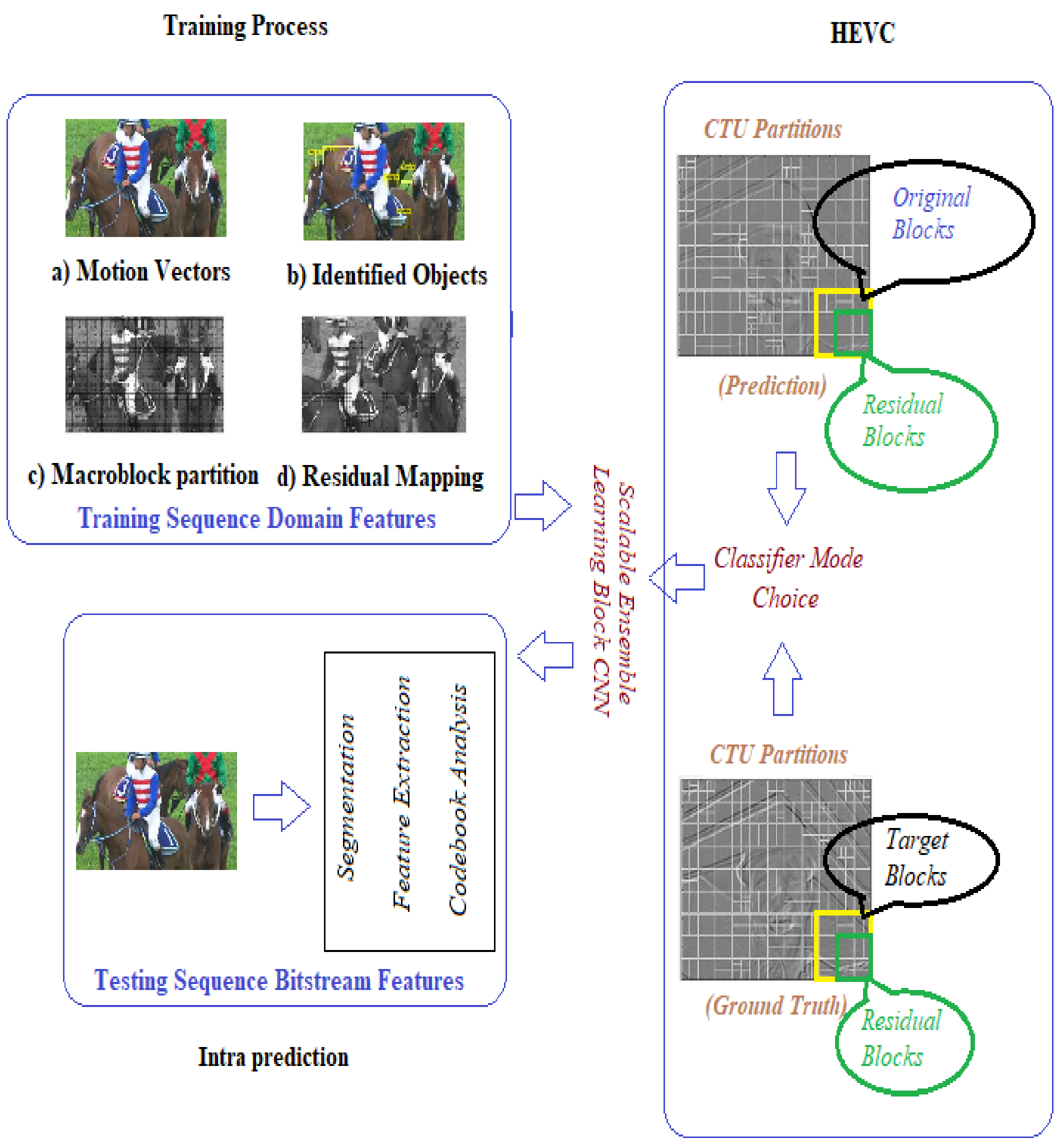


Figure 5.1: Proposed Scalable Ensemble Learning Block CNN (SECNN) for Intra Prediction HEVC.

With these requirements, this paper resorts to a CNN arrangement, which is viewed as fit to manage these difficulties. The benefits are: The CNN's 2D channel is used in a continuous weight combination of multiple spatial areas in the same neighbourhood, which is beneficial for artefact elimination.

Second, for intra ensemble development and reconstruction, the CNN structure connects two modules. Through the final clear image, this structure yields intra ensemble ideal. CNN is more representative than many earlier models, such as pair-wise MRF, and it runs quickly during testing since it is built on few convolution operations.

5.8 Simulations and Results

In order to achieve better performance, the minimum training data size of a PU block is 4x4 and 8x8. However, the prediction of the current PU is generated from the top reference pixels and left reference pixels. A part of the information has deficiency vis-a-vis nearest reference pixels; this is because the nearest reference pixel is weak. Conversely, if the size of the PU block and reference pixel are large, the computational complexity may be better. So, for input training data of CNN, this work chose 16 x 16 which includes a 8 x 8 PU block and its three nearest 8X8 reconstruction blocks. The 8 x 8 PU block is the prediction of the use of the intra-mode in HEVC regular in reference training data. The output data is 16 x 16 block, which is the residual number of input data and original data. Considering as the size of the current PU block is 8x8; the initial values of PU block are expected by the preferred mode in HEVC; and the reconstruction blocks are the references which are probably used for prediction of the PU block.

The size of 16x16 is chosen for prediction for the following three reasons

1. It consists of learning the model by training data of PU block and three nearest reference blocks
2. Improved accuracy of prediction block in HEVC patterns inside the PU block.

3. Making reconstruction processing more accurate at the upper left three reference blocks provided in the reconstruction patterns

In the experiment, the PU block of the training data was used for intra-prediction in HEVC. Besides, using the deep learning framework MatLab for training the network. It is easy to embed the observed frame network into the HEVC reference software. To confirm the general overall performance of the proposed scheme, implement it in reference to HM- 14.0 in MatLab. The learning test sequences encompass a large type of HEVC video sequences. Training datasets are from 10 sequences of four with quantization parameters (QPs): 22, 27, 32 and 37, with only luminance detail is considered. For each QP, a separate network is trained. While comparing with HEVC, the outcomes are evaluated with PSNR, in which the low value indicates bitrate saving and the high value indicates bitrate increase. After assessment with the resource of the use of the PSNR outcomes, moreover study the patterns that the network has observed out and show some figures for details.

In the experiment, a video sequences selected for analysis are given in table 5.1.

Table 5.1: CNN Ensemble learning work Sequence Classes

Class.	**Size.**	**Sequence.**	**No. of Frame.**	**Frame Rate (fps).**
Class A.	(2560X1600).	PeopleStreet.	150.	30.
		Traffic.	500.	50.
Class B.	(1920X1080).	Kimono.	240.	24.
		ParkScene.	500.	50.
Class C.	(832X480).	BQMall.	600.	60.
		PartyScene.	500.	50.

Class D.	(416X240).	BasketballPass.	500.	50.
		RaceHorses.	300.	30.

A frame block considered in the video taken for processing is shown in figure 5.2 Results with PSNR are presented through the proposed work block extraction are in figure 5.2.

Figure.5.2(a): Original extracted Frame

Figure.5.2(b): Identified Objects Frame

Figure.5.2(c): Processing Frame

Figure. 5.2(d): Encoding I frame

Figure. 5.2(e): Encoding B frame

Figure. 5.2(f): Encoding P frame

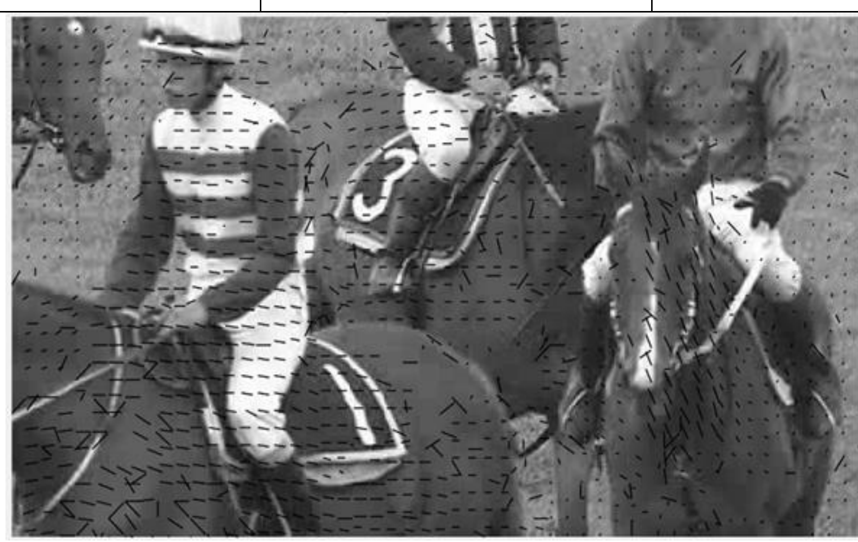


Figure. 5.2(g): Encoding Super imposed Motion Vectors between frame 7 to 16

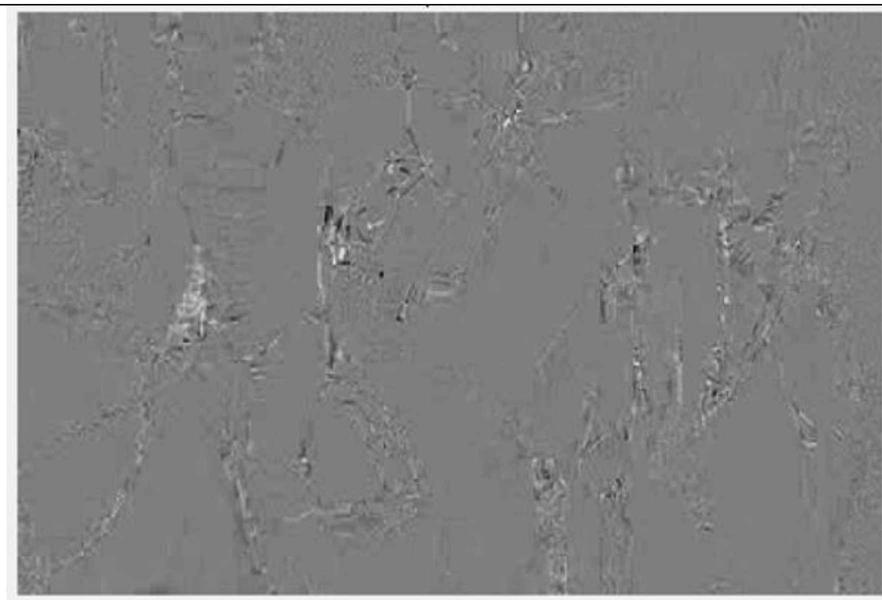


Figure. 5.2(h): Encoding : Stretched Motion Difference for motion compensation prediction between frame 7 to 16

Figure. 5.2: Illustration of Proposed Motion Compensation Frames Extraction (from Figure 5.2 (a) to Figure 5.2 (h))

In the following figure 5.3 and figure 5.4, analyzing the proposed Motion Learning method

Figure. 5.3(a): Motion Learning Frame 7

Figure. 5.3(b): Motion Learning Frame 9

Figure. 5.3(c): Motion Learning Frame 11

Figure. 5.3(d): Motion Learning Frame 13

Figure. 5.3(e): Motion Learning Frame 15

Figure. 5.3: Illustration of Proposed Motion Learning method Frames Extraction (from Figure 5.3 (a) to Figure 5.3 (e))

Figure. 5.4(a): Intra 22 Image

Figure. 5.4(b): Intra 27 Image

Figure. 5.4(c): Intra 32 Image	Figure. 5.4(d): Intra 37 Image
Figure. 5.4: Illustration of Proposed Intra Prediction Mode (22, 27, 32, 37) Frames Extraction (from Figure 5.4 (a) to Figure5.4 (d))	

In the following figure 5.5, Processing Frame Layer wise Learning Features are illustrated

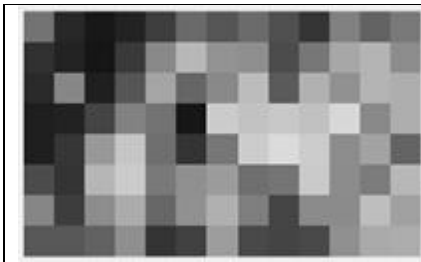
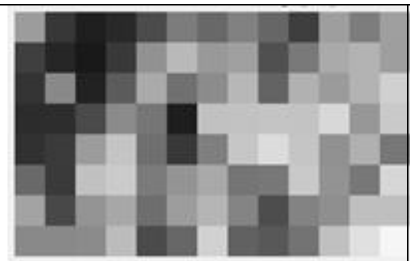
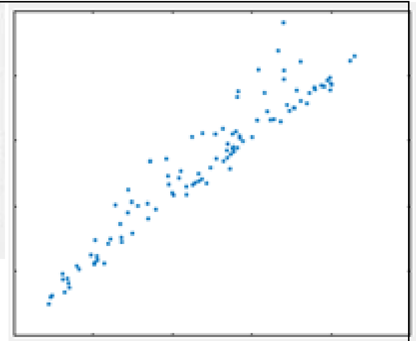
		

Figure. 5.5(a): Mode 22 Actual Learning Layer 1	Figure. 5.5(b): Mode 22 Exact Learning Layer 1	Figure. 5.5(c): Intra CTU Level layer 1.

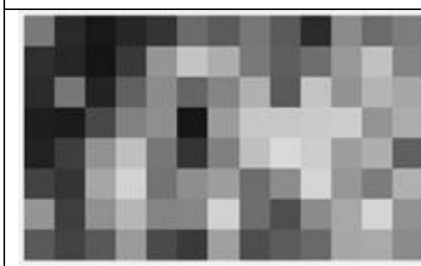
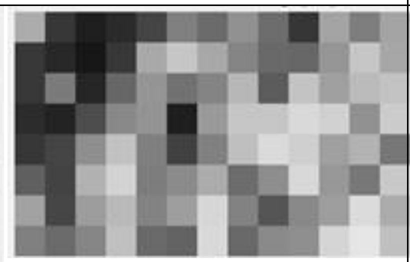
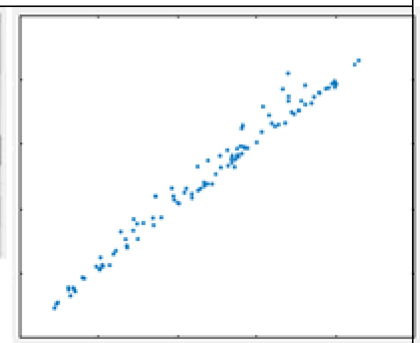

		

Figure. 5.5(d): Mode 27 Actual Learning Layer 2	Figure. 5.5(e): Mode 27 Exact Learning Layer 2	Figure. 5.5(f): Intra CTU Level layer 2.

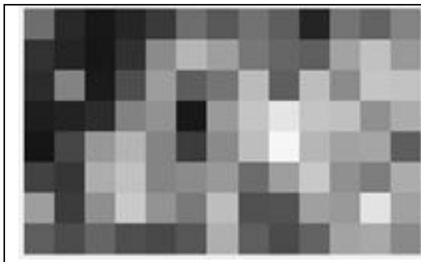
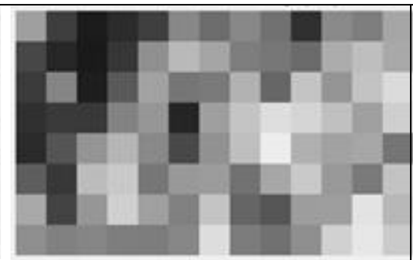
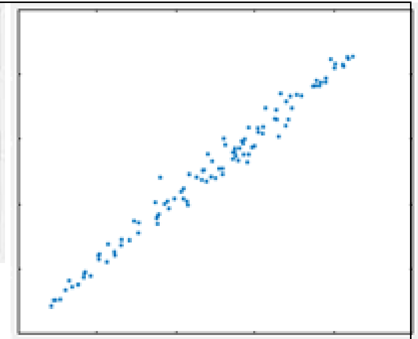
		

Figure. 5.5(g): Mode 32 Actual Learning Layer 3	Figure. 5.5(h): Mode 32 Exact Learning Layer 3	Figure. 5.5(i): Intra CTU Level layer 3.

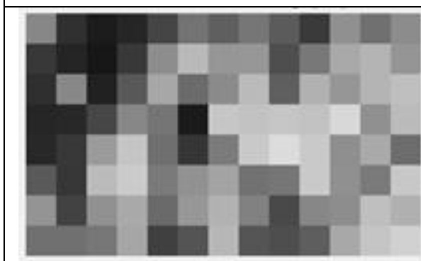
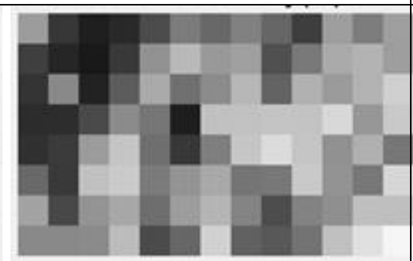
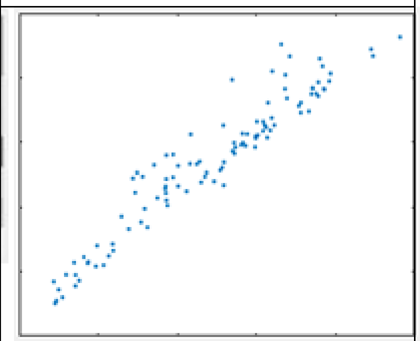

		

Figure. 5.5(j): Mode 37 Actual Learning Layer 4	Figure. 5.5(k): Mode 37 Exact Learning Layer 4	Figure. 5.5(l): Intra CTU Level layer 4.

Figure.5.5: Illustration of Proposed Processing Frame Layer wise Learning Features with Intra CTU Level layer with Intra Prediction Mode (22, 27, 32, 37) Frames Extraction (from Figure 5.5(a) to Figure 5.5(l)) .Here Intra CTU Level layer has taken as estimated values along the x-axis and standard values y-axis through similarity graph.

For various GOP, it might shift significantly. Because of macro-block examination, the actual block frames in the GOP ought to be partitioned by four. The decoded frames are displayed in figure 5.6.

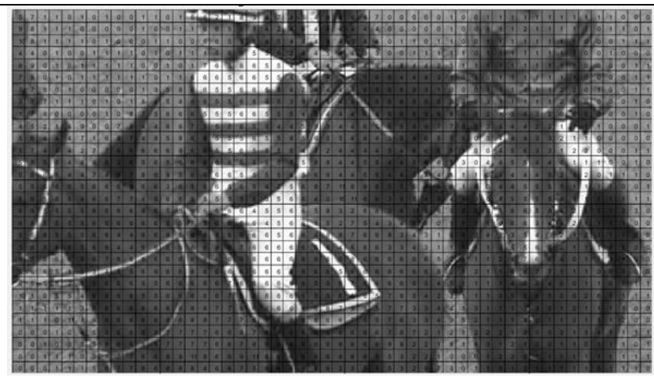


Figure. 5.6(a): Decoding : Magnitude values Ensemble Learning between frame 7 to 16

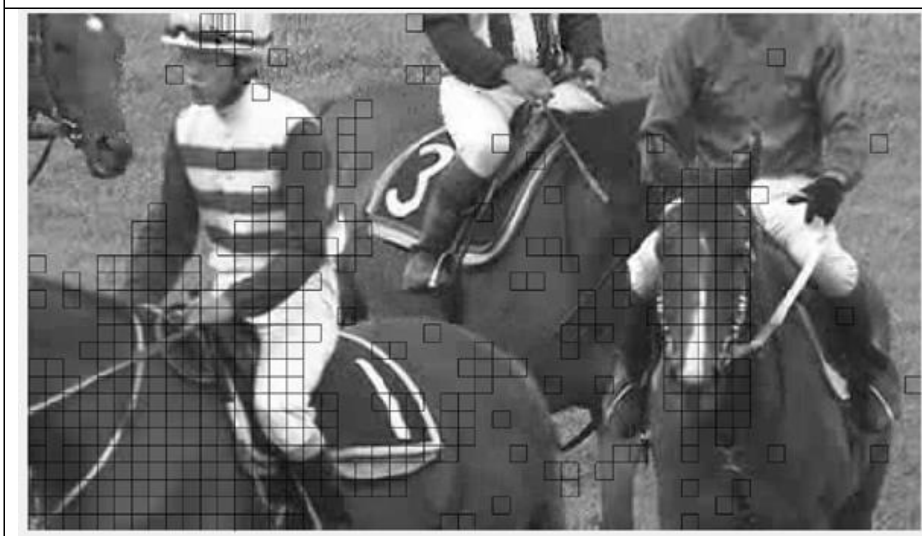


Figure. 5.6(b): Decoding : Selected Motion Vectors for Ensemble Training : Analysis between frame 7 to 16

Figure. 5.6(c): Decoding : Selected Motion Vectors for Ensemble Testing and Extraction : Analysis between frame 7 to 16

Figure. 5.6(d): Decoding Reconstructed Frame

Figure. 5.6: Illustration of Proposed Intra Prediction Mode (22, 27, 32, 37) Decoded Frames Reconstruction (from Figure 5.6 (a) to Figure 5.6 (d))

For various GOPs, it shifts significantly. Because of macro block examination, the actual block frames in the GOP ought to be partitioned by four. Block mode improved and comparative PSNR results are shown below in table 5.2.

Table 5.2: PSNR Comparative Results of Different Surveyed Methods.

Class.	**Sequence.**	**[13].**	**[14].**	**[15].**	**Proposed.**
Class B.					
	Kimono.	39.8	39.8	39.7	39.8
	BQTerrace.	28.5	29.6	30.2	30.2
Class C.					
	BasketballDrill.	31.4	32.7	33.1	33.2
	BQMall.	28.3	29.3	29.4	29.6
Class D.					
	BasketballPass.	30.3	31.3	31.5	31.6
	RaceHorses.	29.5	31.3	31.5	31.6

5.9 Summary

In this chapter the proposed algorithm was simulated and tested against various video sequences as shown in table 5.2. In all these sequences the implemented intra prediction algorithm shows better PSNR with comparison to existing algorithms as shown in table 5.2.

CHAPTER-6

A Neural Network-based Inter-frame Prediction for High Efficiency Video Coding

This chapter presents details of recurrent neural network, its limitations and performance improvements of recurrent neural network. It also gives information about long short term memory and its different gates. Finally it discusses the proposed method for intra frame prediction of high efficiency video codec and simulation results.

6.1 Introduction

Video compression and decompression or codec algorithms have been around for four decades. They have become a necessity in today's era, owing to the ever- increasing resolution capabilities of video cameras and their increasingly high storage and transmission requirements. It can be seen that while an average 1080p video of length 1 minute would turn out to be approximately 130MB in size, the needed bandwidth to stream such videos is only between 8 to 16 Mbps. This can be attributed to advancements in video codec algorithms. The High-Efficiency Video Coding standard/ H.265 is one of the latest standards in video encoding designed by MPEG as a successor to the H.264/ AVC standard. It promises a 25 percent to 50 percent reduction in bit rate without compromise in video quality. Though similar in architecture, the improved features in the H.265 include the use of CTUs instead of macroblocks, better in-loop filters, etc., leading to greater accuracy in the encoding process with a reduction in bit rate.

A prominent characteristic of video data is redundancy. This basically refers to the similarities within video data. There are two types of redundancies: spatial, referring to the similarities within a specific image, and temporal, referring to the similarities between two consecutive frames. Using these redundancies to obtain residual frames, which are much lesser in size when compared to the original frames, is a pivotal part of video compression. Lesser the error between a reconstructed frame and the original frame, the lesser the data that is to be encoded. In this paper, the aim is to use an LSTM or Long Short Term based deep learning approach to do inter- frame prediction to come up with a reconstructed frame, which has to be made as accurate as possible. This is done with a sequence of N previous frames to obtain a predicted (N+1)th frame, with which a residual frame is generated, occupying much less space when encoded. The paper is structured as follows: an overview of HEVC and LSTMs is provided, followed by the construction of the neural network architecture for inter-frame prediction, concluding the same with results and comparisons.

6.2 Recurrent Neural Networks

Consider the stock market data of a certain stock as an example of sequential data. Based on the number of features, a machine learning mode or AI predicts the stock prices, stock volume, Value of Opening, and so on. Because stock price is dependent on these features, it is also primarily reliant on previous day's values of the stock. In truth, the value of the previous day or days is one of the most essential deciding criteria for a trader.This dependency on time is achieved by RNN.RNN model block diagram is shown below.

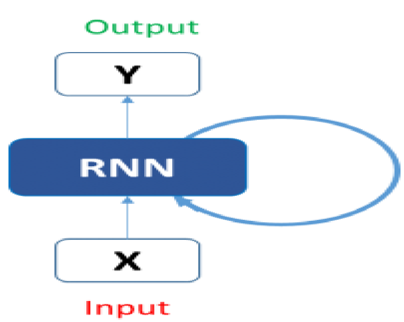


Figure 6.1 Block diagram of RNN

The simple unfolded diagram of RNN is shown below.

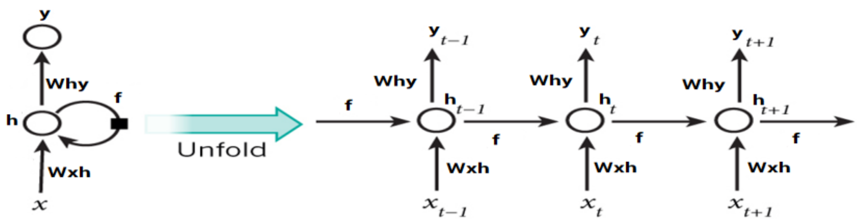


Figure 6.2 Unfolded diagram of RNN

It's now simple to see how these networks interpret stock price trends and predict the stock prices of the day. Each prediction at a specific time t (h_t) is not only based on past days predictions, but also on the knowledge learned for them. To a large extent, but not totally, recurrent neural networks can address our sequence handling problem. For example, we'd like our computer to compose Shakespearean poetry. RNNs are fantastic for short contexts, but to tell a tale and remember it, we need RNN models that can grasp and recall the context behind the sequences in the same way that a human brain can. With a simple RNN, this is impossible.

6.3 Limitations of Recurrent Neural Network

RNNs are excellent at coping with short-term dependency consider simple prediction problem shown below [118].

The colour of the sky is ____

The RNN isn't interested in what happened or what it meant earlier during prediction all it cares about is that the sky is mostly blue. The RNN prediction is shown below as a result.

The colour of the sky is blue.

In contrast, RNNs are unable to recognize the context after an input. It's hard to recall what was said in the past when making forecasts in the present. Let’s consider an example shown below.

I spent 20 long years working for the under-privileged kids in Spain. I then moved to Africa.

……….

I can speak fluent ________.

We may conclude from the following that the author has a solid grasp of the language, having worked in Spain for 20 years. Recurrent neural networks, on the other hand, need to remember this context in order to generate better predictions. A considerable quantity of irrelevant data may exist between the relevant data and the point when it is needed. In this situation, RNN fails.

The Vanishing Gradient problem is thus at the core of the issue.To avoid vanishing gradient problem feed-forward RNN will be used, before one can comprehend how a feed-forward RNN learns, one must first understand how a feed-forward RNN works. The weight update sent to a single layer in a traditional feed-forward neural network is a function of learning rate, error term from the previous layer, and input to that layer. As a result, each layer's error term is simply the accumulation of errors from previous layers. As the early layers deepen, the modest values of the derivatives of activation functions like the sigmoid are amplified many times. As a result, as we get closer to the original layers, the gradient almost disappears.

RNN is in a similar situation. Recurrent Neural Networks only store information for short periods of time, so if we need the information in a hurry, we can get it, but once a big quantity of words is fed in, the knowledge is gone. A modified variant of RNN called as LSTM can be used to tackle the difficulty outlined above.

6.4 Improvement over Recurrent Neural Network

When we prepare our day's itinerary, we prioritise our appointments. We know which meeting might have to be cancelled to make room for something more important. It makes use of a function that completely changes the present data in order to include new data. As a result, all of the data has been changed, with no differentiation made between 'important' and 'less significant' material. Long Term Short Memory perform little changes to the data using multiplications and additions. In Long Term Short Memory, cell states are a way of communicating information.

The likeness of LSTM to conveyor belts is another distinctive feature. In industries, they use conveyor belts to to move goods for numerous activities. LSTMs employ this strategy to shuffle data around. Information can be added, updated, or withdrawn as it progresses through the steps, just like a product on a conveyor line might be moulded, painted, or packed.

The close interaction between LSTMs and conveyor belts is seen in the diagram below.

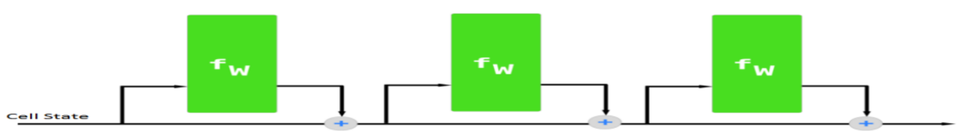


Figure 6.3 Conveyor belt of LSTM

Though the figure does not accurately depict the real architecture of an LSTM, the companson is available to get an idea of LSTM architecture and function. They may forget and recall things because of an LSTM characteristic that allows them to selectively forget and recollect information by tweaking it little by little rather than changing the complete information.

6.5 Architecture of Long Short Term Memory

LSTMs (Long Short Term Memory) come under a class of neural networks called recurrent neural networks or RNNs. The various neurons of a recurrent neural network have an internal neuron state, which serves as a memory. This neuron state is used to process incoming information to the neural network. However, this internal memory is not stored for very long in a simple RNN, which in turn led to the formation of LSTMs.

LSTMs are capable of learning short-term as well as long-term correlations in the incoming data. This is done through the help of three different internal layers. Each of these layers helps in obtaining data, selective learning or forgetting data, and providing an output to the next layer. The sigmoid and tanh activation functions are crucial in an LSTM. The input layer obtains information from the previous time step, the forget layer decides which information has to be retained and which information to be forgotten, and the output layer decides which information should go to the next layer of the LSTM network. This is very useful when dealing with temporal data and time series.

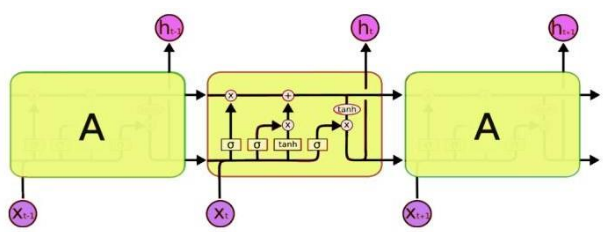


Figure 6.4: An overview of an LSTM network.

While dealing with image data, instead of normal matrix multiplications in neural networks, convolution operations are performed for better feature learning by the neural network. Hence, instead of a normal LSTM, a convolutional LSTM is used for the task of inter-frame prediction.

6.5.1 Forget Gate of LSTM

The process of forgetting the topic is controlled by the forget gate. Below figure shows forget gate of LSTM.

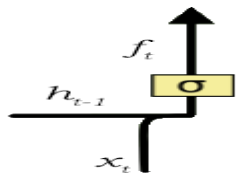


Figure 6.5: Forget gate of LSTM

The input at that time step is x $_t$, and h $_{t-1}$ is the hidden state from the preceding cell or its output. The weight matrices are multiplied by the inputs, and a bias is then imposed. Then, this value is subjected to a sigmoid function. A vector of 0 to 1 values, one for each cell state number, is the result of the sigmoid function. Which data should be saved and which should be discarded is decided using the sigmoid function. The forget gate wants the cell state to entirely forget about a certain value when it outputs a '0' for that value in the cell state. In the same way, 1 means that the forget gate is open.

6.5.2 Input Gate of LSTM

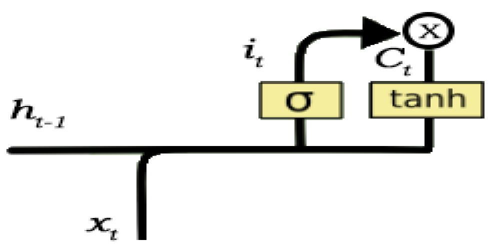


Figure 6.6: Input gate of LSTM

The input gate is responsible for incorporating data into the cell's current state. The process of adding information consists of three steps, as seen in the image above. A sigmoid function regulates how values are added to the cell state. Similar to how the forget gate filters all data from h_{t-1} and x_t, this gate also filters all data from the two sources.

Constructing a vector that contains every possible value that could be added to the cell state (as determined by h_{t-1} and x_t), the tanh function, which returns values between -1 and +1, is used to do this. By dividing the value of the regulatory filter (the sigmoid gate) by the generated vector, this crucial information may be added to the cell state (the tanh function). We ensure that only significant and non-redundant information is introduced to the cell state once this three-step process is done.

6.5.3 Output Gate of LSTM

Consider the following scenario as an example for output gate [118].

Bob fought single handedly with the enemy and died for his country. For his contributions brave _____.

The blank space in this statement might be filled with a variety of options. The current input, 'brave,' is an adjective that characterizes a noun, as we already know. As a result, the following word will almost certainly be a noun. As a consequence, Bob may be a fantastic result. The output gate's task is to extract useful information from the current cell state and provide it to the user. The following is a description of how it works.

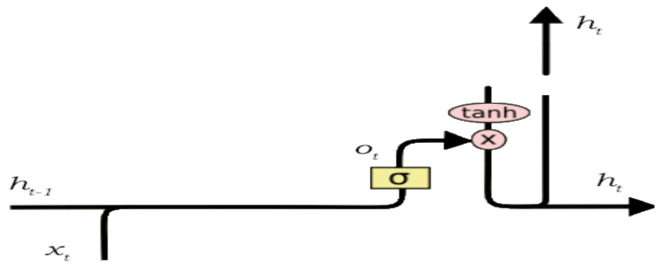


Figure 6.7: Output gate of LSTM

The function of an output gate may be broken down into three pieces:

1) After applying the tanh function on the cell state and generating a vector, the values are scaled to the range -1 to +1.

2) Use the h $_{t-1}$ and x $_t$ values to establish a filter to control the values that must be formed from the vector generated before. The sigmoid function is used once more in this filter.

3) Multiplying the value of this regulatory filter by the vector produced in step 1, then sending the result as an output as well as to the hidden state of the next cell. If 'Bob' is excluded, the filter in the preceding example will lower all other values. As a result, the filter must be constructed and applied to the cell state vector using the input and concealed state values.

6.6 Simulations and Results

LSTM Based Inter Frame Prediction Results

Inter-frame prediction was performed on a number of films with varied ranges of motion. Below are the video sequence, ground truth, and forecasts.

Figure 6.8: The simulation results are listed in the following order: picture sequence, ground truth, and forecast frame.

Each sequence's residual frames are listed below.

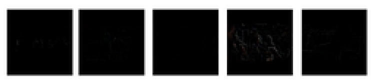


Figure.6.9: For the 6th picture of each series, residual frames were created.

A. PSNR Calculations

In terms of PSNR, a conventional H.265 motion vector-based inter-frame prediction is compared to a neural network-based prediction.

Table 6.1: PSNRs of a traditional motion compensation-based inter-frame prediction and our neural network-based prediction are compared.

Image Sequence	**PSNR of Motion vector-based inter-frame prediction**	**PSNR of Neural Network-based inter-frame prediction.**
Basketball	33.15	38.91
Blowing Bubbles	34.23	39.931
Market	33.15	44.809
Party	33.20	25.869
Man Walking	33.18	38.678

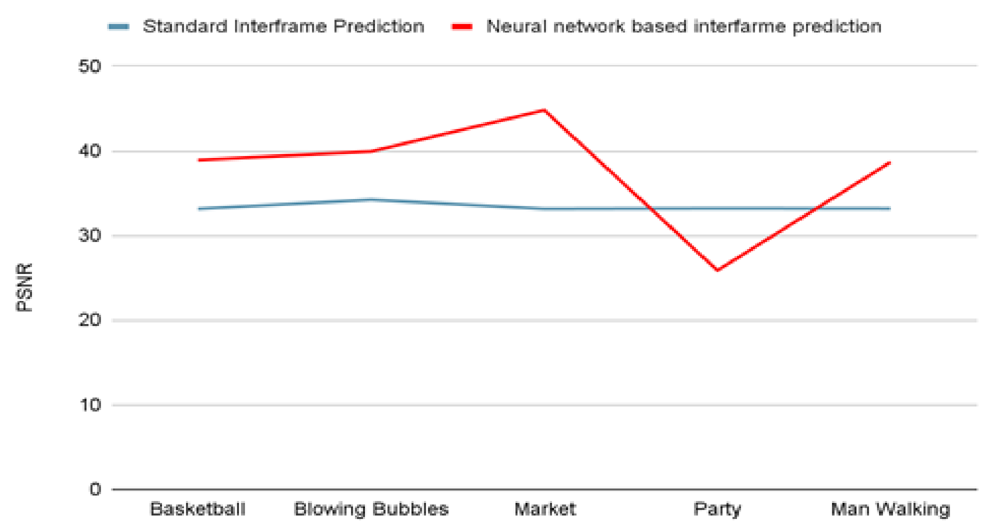


Figure.6.10: A graphical representation of PSNR comparison of the two methods

SSIM Calculations

Unlike PSNR, SSIM is concerned with perceived similarity between two pictures. It calculates how similar the two photos are visually.

Table. 6.2: The SSIMs of our neural network-based prediction and a classic motion compensation-based inter-frame prediction are compared.

Image Sequence	**SSIM of Motion vector-based inter-frame prediction**	**SSIM of Neural Network-based inter-frame prediction.**
Basketball	97.38%	99.96 %
Blowing bubbles	98.89%	99.99%
Market	99.91%	99.99%
Party	98.59 %	97.63%
Man Walking	99.72%	99.99%

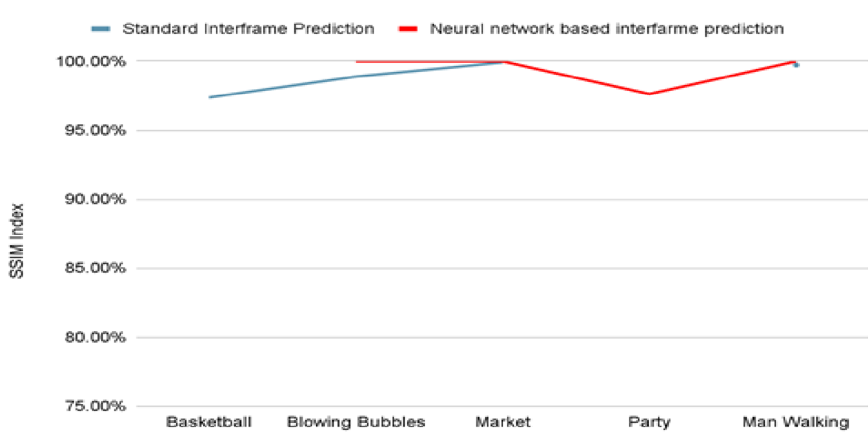


Figure 6.11: Comparing SSIMs of a standard motion compensation-based inter-frame predict.

From the above columns, one can find that the average PSNR is 37.53 dB, and the average SSIM is 99.518%. One can also find an improvement of 11 percent in PSNR and percent 0.6 in SSIM.

6.7 Summary

With the advent of powerful processors, training and implementation of complex neural networks for such information-intensive processes are no longer out of our hands. With a neural network-based implementation, one can see the better accuracy of the inter-frame prediction module when compared to a more primitive algorithmic approach. Further improvements in compression technology will enable us to transmit and store very high- quality video (8K) in much lower spaces than required

CHAPTER-7

Conclusion and Scope for the Future Research

The thesis provides an image resolution predicted-ensemble system that uses CNN to solve the intra frame prediction problem. It has been shown that image resolution prediction obtained using simple feed forward reproduction algorithms applying a fluctuating motion estimate setting typically contains appropriate data for evaluating the final image resolution picture. The use of CNN to introduce the reconstruction and de-convolution processes is made in this research. On several intra prediction frames, the proposed technique gives better results. This study makes use of an Ensemble-based optimum bit allocation (OBA) and rate-distortion optimization (RDO) for HEVC. To achieve this goal, a precise CTU-level intra prediction model is developed primarily, which has been found to be more accurate than two widely used models. The Ensemble-based RDO may be accomplished using this model, which is based on a low-resolution picture and an image comparability-related Lagrangian multiplier derived by Ensemble-based OBA. Furthermore, greater magnitude prediction and motion vectors were also achieved, which is equal to content information primarily in content district regions and at the edges; for smooth areas, the adjustment was little or even negligible, resulting in high imperceptibility. Future work need to focus on improving the structure's ability to handle considerably greater super resolution proportions and incorporating picture resolution prediction generation into CNN network.

We can now train and use complex neural networks for such information-intensive tasks because of the development of powerful computers. When compared to a more simple algorithmic approach, the inter-frame prediction module's accuracy is shown to be higher with neural network-based implementation. We will be able to broadcast and store extremely high-quality video (8K) in considerably less space as compression technology advances with time.

References

[1] A convolutional neural network approach for post-processing in HEVC intra coding (MMM 2017), Y. Dai, D. Liu, F. Wu.

[2] Decoder-Side Hevc Quality Enhancement With Scalable Convolutional Neural Network (ICME 2017), Ren Yang, Mai Xu and Zulin Wang.

[3] Enhancing Quality for HEVC Compressed Videos (Arxiv 2017), Ren Yang, Mai Xu, Tie Liu, Zulin Wang, and Zhenyu Guan

[4] Cnn Based Post-Processing To Improve Hevc (ICIP 2017), Chen Li, Li Song, Rong Xie and Wenjun Zhang.

[5] The Multi-Scale Deep Decoder for the Standard HEVC Bitstreams (DCC 2018), Tingting Wang, Wenhui Xiao, Minglin Chen and Hongyang Chao.

[6] Enhancing Hevc Compressed Videos With A Partition-Masked Convolutional Neural Network (ICIP 2018), Xiaoyi He, Qiang Hu, Xintong Han, Xia.

[7] Neural network-based arithmetic coding of intra prediction modes in HEVC (VCIP 2017), R. Song, D. Liu, H. Li, F. Wu.

[8] A Convolutional Neural Network Approach for Post-Processing in HEVC Intra Coding (MMM 2017), Yuanying Dai, Dong Liu, and Feng Wu.

[9] Convolutional Neural Networks based Intra Prediction for HEVC (Arxiv 2018), Wenxue Cui, Tao Zhang, Shengping Zhang, Feng Jiang, Wangmeng Zuo, Debin Zhao.

[10] CU partition mode decision for HEVC hardwired intra encoder using convolution neural network (TIP 2016), Z. Liu, X. Yu, Y. Gao, S. Chen, X. Ji, D. Wan.

[11] Learning Binary Residual Representations for Domain-specific Video Streaming (AAAI 2018), Y.-H. Tsai, M.-Y. Liu, D. Sun, M.-H. Yang, J. Kautz.

[12] Video compression through image interpolation (Arxiv 2018), C.-Y. Wu, N. Singhal, P. Krähenbühl.

[13] Convolutional Neural Network-Based Block Up-sampling for Intra Frame Coding (TCSVT 2017), Yue Li, Dong Liu, Houqiang Li, Li Li, and Feng Wu

[14] Variable Block-Sized Signal Dependent Transform for Video Coding (TCSVT 2017), Cuiling Lan, Jizheng Xu, Wenjun Zeng, Guangming Shi, Feng Wu.

[15] Reinforcement Learning for HEVC/H.265 Intra-Frame Rate Control (ISCAS 2018), Jun-Hao Hu, Wen-Hsiao Peng, and Chia-Hua Chung.

[16] Neural Network Based Inter Prediction for HEVC (ICME 2018), Yang Wang, Xiaopeng Fan, Chuanmin Jia, Debin Zhao, Wen Gao.

[17] Enhanced Image Decoding via Edge-Preserving Generative Adversarial Network (ICME 2018), Qi Mao, Shiqi Wang, Shanshe Wang, Xinfeng Zhang, Siwei Ma.

[18] Fully Connected Network for HEVC CU Split Decision equipped with Laplacian Transparent Composite Model (PCS 2018), Hossam Amer, Abdullah Rashwan, En-hui Yang.

[19] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, “Overview of the high efficiency video coding (HEVC) standard,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, no. 12, 2012.

[20] M. Wang, K. N. Ngan, and H. Li, “An efficient frame-content based intra frame rate control for high efficiency video coding,” IEEE Signal Processing Letters, vol. 22, no. 7, pp. 896–900, 2015.

[21] M. Wien, “High efficiency video coding: coding tools and specification,” Springer, September 2014.

[22] J. Lainema, F. Bossen, W.-J. Han, J. Min, and K. Ugur, “Intra coding of the HEVC standard,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, no. 12, pp. 1792–1801, December 2012.

[23] A. J. Bell and T. J. Sejnowski, “Edges are the independent components of natural scenes,” in NIPS, 1996.

[24] J. H. van Hateren and A. van der Schaaf, “Independent component filters of natural images compared with simple cells in primary visual cortex,” Proc. R. Soc. Lond. B, vol. 265, no. 1394, pp. 359–366, March 1998.

[25] H. Lee, C. Ekanadham, and A. Y. Ng, “Sparse deep belief net model for visual area V2,” in NIPS, 2007.

[26] C. Dong, Y. Deng, C. C. Loy, and X. Tang “Compression Artifacts Reduction by a Deep Convolutional Network” IEEE International Conference on Computer Vision (ICCV) 2015.

[27] J. Kim, J. Lee, and K. M. Lee "Deeply-recursive convolutional network for image super-resolution." in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016

[28] W. S. Park and M. Kim, "CNN-based in-loop filtering for coding efficiency improvement." IEEE 12th Image, Video, and Multidimensional Signal Processing Workshop (IVMSP), 2016

[29] Kang, Jihong, Sungjei Kim, and Kyoung Mu Lee. "Multi-modal/Multiscale convolution neural network based in-loop filter design for next generation video codec." in Proceedings of the International Conference on Image Processing, 2017.

[30] Li, Y., Liu, D., Li, H., Li, L., Wu, F., Zhang, H. and Yang, H., "Convolutional Neural Network-Based Block Up-sampling for Intra Frame Coding"IEEE Transactions on Circuits and Systems for Video Technology (2017).

[31] J. Li, B. Li, J. Xu and R. Xiong, "Intra prediction using fully connected network for video coding," IEEE International Conference on Image Processing (ICIP), 2017.

[32] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," IEEE Trans. Pattern Anal. Mach. Intell., vol. 38, no. 2, pp. 295–307, 2016.

[33] C. Dong, Y. Deng, C. C. Loy, and X. Tang, "Compression artifacts reduction by a deep convolutional network," Proc. IEEE Int. Conf. Computer Vision, pp. 576–584, 2015.

[34] Y. Dai, D. Liu, and F. Wu, "A convolutional neural network approach for post-processing in HEVC intra coding," Proc. Int. Conf. Multimedia Modeling, pp. 28–39, 2017.

[35] W. S. Park and M. Kim, "CNN-based in-loop filtering for coding efficiency improvement," Proc. IEEE Image, Video, and Multidimensional Signal Processing Workshop, July 2016.

[36] J. Li, B. Li, J. Xu, and R. Xiong, "Intra prediction using fully connected network for video coding," in Proceedings of ICIP 2017, 2017.

[37] W. Cui, T. Zhang, S. Zhang, F. Jiang, W. Zuo, Z. Wan, and D. Zhao, "Convolutional neural networks based intra prediction for HEVC," in Data Compression Conference (DCC), 2017, IEEE, 2017, pp. 436–436.

[38] I Krasin, T Duerig, N Alldrin, A Veit, S Abu-El-Haija, S Belongie, D Cai, Z Feng, V Ferrari, V Gomes, et al., "Openimages: A public dataset for large-scale multi-label and multi-class image classification," Dataset available from https://github. com/openimages, vol. 3, 2016.

[39] HEVC test model (HM). [Online]. Available: https://hevc.hhi. fraunhofer.de/.

[40] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. v.d. Smagt, D. Cremers, and T. Brox. "Flownet:Learning optical flow with convolutional networks". In ICCV,2015

[41] Liu, Z., Yeh, R., Tang, X., Liu, Y. and Agarwala, A. "Video frame synthesis using deep voxel flow." International Conference on Computer Vision (ICCV). Vol. 2. 2017.

[42] Niklaus, Simon, Long Mai, and Feng Liu. "Video frame interpolation via adaptive separable convolution." arXiv preprint arXiv:1708.01692 (2017).

[43] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. MIT Press, 2016.

[44] K. K. Chua and Y. H. Tay, "Enhanced image super-resolution technique using convolutional neural network," in Advances in Visual Informatics, 2013, pp. 157–164.

[45] D. Eigen, D. Krishnan, and R. Fergus, "Restoring an image taken through a window covered with dirt or rain," in 2013 IEEE International Conference on Computer Vision, 2013, pp. 633–640.

[46] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2, 2014, pp. 2366–2374.

[47] T. Li, M. Xu, and X. Deng, "A deep convolutional neural network approach for complexity reduction on intra-mode HEVC," in 2017 IEEE International Conference on Multimedia and Expo (ICME), 2017.

[48] T. Laude and J. Ostermann, "Deep learning-based intra prediction mode decision for HEVC," in 2016 Picture Coding Symposium (PCS), 2016.

[49] R. Song, D. Liu, H. Li, and F. Wu, "Neural network-based arithmetic coding of intra prediction modes in HEVC," in 2017 IEEE Visual Communications and Image Processing (VCIP), 2017.

[50] B. Xu, X. Pan, Y. Zhou, Y. Li, D. Yang, and Z. Chen, "CNNbased rate-distortion modeling for H.265/HEVC," in 2017 IEEE Visual Communications and Image Processing (VCIP), 2017.

[51] L. Zhou, X. Song, J. Yao, L. Wang, and F. Chen, "JVET-IO022-v3: Convolutional neural network filter (CNNF) for intra frame," Tech. Rep., 2018.

[52] Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian, "Pointwise shape-adaptive DCT for high-quality denoising and deblocking of grayscale and color images," IEEE TIP, 2007.

[53] Ci Wang, Jun Zhou, and Shu Liu, "Adaptive non-local means filter for image deblocking," Signal Processing: Image Communication, pp. 522–530, 2013.

[54] Jeremy Jancsary, Sebastian Nowozin, and Carsten Rother, "Lossspecific training of non-parametric image restoration models: A new state of the art," in ECCV, 2012.

[55] Huibin Chang, Michael K Ng, and Tieyong Zeng, "Reducing artifacts in JPEG decompression via a learned dictionary," IEEE TSP, pp. 718– 728, 2014.

[56] Cheolkon Jung, Licheng Jiao, Hongtao Qi, and Tian Sun, "Image deblocking via sparse representation," Signal Processing: Image Communication, pp. 663–677, 2012.

[57] Yann LeCun, L´eon Bottou, Yoshua Bengio, and Patrick Haffner, "Gradient-based learning applied to document recognition," Proceedings of the IEEE, pp. 2278–2324, 1998.

[58] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," in NIPS, 2012.

[59] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei, "Large-scale video classification with convolutional neural networks," in CVPR, 2014.

[60] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in CVPR, 2014.

[61] Jonathan Long, Evan Shelhamer, and Trevor Darrell, "Fully convolutional networks for semantic segmentation," in CVPR, 2015.

[62] Chao Dong, Yubin Deng, Chen Change Loy, and Xiaoou Tang, "Compression artifacts reduction by a deep convolutional network," in ICCV, 2015.

[63] Zhangyang Wang, Ding Liu, Shiyu Chang, Qing Ling, and Thomas S Huang, “D3: Deep dual-domain based fast restoration of jpegcompressed images,” arXiv preprint arXiv:1601.04149, 2016.

[64] Qinglong Han and Wai-Kuen Cham, “High performance loop filter for HEVC,” in ICIP, 2015.

[65] Woon-Sung Park and Munchurl Kim, “CNN-based in-loop filtering for coding efficiency improvement,” in IVMSP. IEEE, 2016.

[66] Yuanying Dai, Dong Liu, and Feng Wu, “A convolutional neural network approach for post-processing in HEVC intra coding,” arXiv preprint arXiv:1608.06690, 2016.

[67] M. Elad and Y. Hel-Or. A fast super-resolution reconstruction algorithm for pure translational motion and common space-invariant blur. IEEE TIP, 10(8):1187–1193, 2001.

[68] S. Kim, N. K. Bose, and H. Valenzuela. Recursive reconstruction of high resolution image from noisy undersampled multiframes. IEEE Transactions on Acoustics, Speech and Signal Processing, 38(6):1013–1027, 1990.

[69] C. Liu and D. Sun. On bayesian adaptive video super resolution. IEEE TPAMI, 2013.

[70] H. Takeda, P. Milanfar, M. Protter, and M. Elad. Superresolution without explicit subpixel motion estimation. IEEE TIP, 18(9):1958–1975, 2009.

[71] H. Ur and D. Gross. Improved resolution from subpixel shifted pictures. CVGIP, 54(2):181–186, 1992.

[72] S. Ma, S. Wang, S. Wang, L. Zhao, Q. Yu, and W. Gao, “Low complexity rate distortion optimization for HEVC,” in Proc. Data Compression Conference (DCC), Mar. 2013, pp. 73-82.

[73] J. Zhu, Z. Liu, D. Wang, Q. Han, and Y. Song, “Fast prediction mode decision with hadamard transform based rate-distortion cost estimation for HEVC intra coding,” in Proc. IEEE International Conference on Image Processing(ICIP), Sep. 2013, pp. 1977-1981.

[74] Z. Liu, S. Guo, and D. Wang, “Binary classification based linear rate estimation model for HEVC RDO,” in Proc. IEEE International Conference on Image Processing(ICIP), Sep. 2014, pp.3676-3680.

[75] H. Zhang and Z. Ma, “Fast intra mode decision for High Efficiency Video Coding (HEVC),” IEEE Trans. Circuits and Systems for Video Technology, vol. 24, pp. 660-668, Nov. 2012.

[76] Y. Zhang, Z. Li, and B. Li, “Gradient-based fast decision for intra prediction in HEVC,” in Proc. Visual Communications and Image Processing (VCIP), Jan. 2012, pp. 1-6.

[77] B. Min and R. C. C. Cheung, “A fast cu size decision algorithm for the HEVC intra encoder,” IEEE Trans Circuits and Systems for Video Technology, vol. 25, pp. 892-896, Oct. 2015.

[78] N. Hu and E.-H. Yang, “Fast mode selection for HEVC intra frame coding with entropy coding refinement based on transparent composite model,” IEEE Trans. Circuits and Systems for Video Technology, vol. 25, pp. 1521-1532, Jan. 2015.

[79] X. Yu, Z. Liu, J. Liu, Y. Gao, and D. Wang, “VLSI friendly fast CU/PU mode decision for HEVC intra encoding: Leveraging convolution neural network,” in Proc. IEEE International Conference on Image Processing (ICIP), Sept. 2015, pp. 1285-1289.

[80] Z. Liu, X. Yu, S. Chen, D. Wang, “CNN oriented fast HEVC intra CU mode decision,” in Proc. IEEE International Symposium on Circuits and Systems (ISCAS), Aug. 2016, pp. 2270-2273.

[81] J. Li, B. Li, J. Xu, R. Xiong, and W. Gao, “Fully-connected network based intra prediction for image coding,” IEEE Transactions on Image Processing, vol. 27, no. 7, pp. 3236–3247, July 2018.

[82] Yu X, Liu Z, Liu J, et al. VLSI friendly fast CU/PU mode decision for HEVC intra encoding: Leveraging convolution neural network[C]//Image Processing (ICIP), 2015 IEEE International Conference on. IEEE, 2015: 1285-1289.

[83] Park W S, Kim M. CNN-based in-loop filtering for coding efficiency improvement[C]//Image, Video, and Multidimensional Signal Processing Workshop (IVMSP), 2016 IEEE 12th. IEEE, 2016: 1-5.

[84] Dai Y, Liu D, Wu F. A Convolutional Neural Network Approach for Post-Processing in HEVC Intra Coding[J]. arXiv preprint arXiv:1608.06690, 2016.

[85] Bjontegaard G. Calcuation of average PSNR differences between RD-curves[J]. Doc. VCEG-M33 ITU-T Q6/16, Austin, TX, USA, 2-4 April 2001, 2001.

[86] J. Agbinya, “Interpolation using the discrete cosine transform,” Electronics Letters, vol. 28, no. 20, pp. 1927–1928, 1992.

[87] J. Kim, J. K. Lee, and K. M. Lee, “Accurate image super-resolution using very deep convolutional networks,” in IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 1646–1654.

[88] Y. Li et al., "Convolutional Neural Network-Based Block Up-Sampling for Intra Frame Coding," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 28, no. 9, pp. 2316-2330, Sept. 2018, doi: 10.1109/TCSVT.2017.27276.

[89] Cho, S & Kim, M 2013, ‘Fast CU splitting and pruning for suboptimal CU partitioning in HEVC Intra- coding’, IEEE Transactions on Circuits and Systems for Video Technology, vol. 23, no. 9, pp. 1555-1564.

[90] Jain, HB & Rao, KR 2014, ‘Fast Intra- mode decision in high efficiency video coding’, Polibits, vol. 50, pp. 5-12.

[91] Gan, Y, Zhao, X & Zhang, Q 2015, ‘Fast algorithm with early termination CU split and mode decision’, Inter-national Journal of Control and Automation, vol. 8, no. 2, pp. 83-94.

[92] Ding, H, Wang, F, Zhang, W & Zhang, Q 2016, ‘Adaptive motion search range adjustment algorithm for HEVC Inter- coding’, Optik- Inter-national Journal for Light and Electron Optics, vol. 127, no. 19, pp. 7498-7506.

[93] Podder, PK, Paul, M & Murshed, M 2016, ‘Fast mode decision in the HEVC video coding standard by exploiting region with dominated motion and saliency features’, PloS One, vol. 11, no. 3, p. e0150673.

[94] Karwowski, D & DomaĹ, M 2016, ‘Context-adaptive binary arithmetic coding with precise probability estimation and complexity scalability for high-efficiency video coding’, Journal of Electronic Imaging, vol. 25, no. 1, p. 013010.

[95] Elyousfi, A 2014, ‘An improved fast mode decision method for H. 264/AVC Intra-coding’, Advances in Multimedia, vol. 2014, pp. 1-8.

[96] Zhe Xu, Biao Min, Ray C.C. Cheung 2018, ‘A fast inter CU decision algorithm for HEVC, Signal Processing: image Communication and Specification, vol.60, pp. 211-223.

[97] Ding, H, Wang, F, Zhang, W & Zhang, Q 2016, 'Adaptive motion search range adjustment algorithm for HEVC Inter- coding', Optik- Inter-national Journal for Light and Electron Optics, vol. 127, no. 19, pp. 7498-7506.

[98] Kerr, DA 2012, 'Chrominance subsampling in digital images', The Pumpkin, vol. 2012, no. 3, pp. 1-15.

[99] Ohm, JR, Sullivan, GJ, Schwarz, H, Tan, TK & Wiegand, T 2012, 'Comparison of the coding efficiency of video coding standards— including High Efficiency Video Coding (HEVC)', IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, no. 12, pp. 1669-1684.

[100] Lee, A, Jun, D, Kim, J, Choi, JS & Kim, J 2014, 'Efficient Inter- prediction mode decision method for fast motion estimation in high efficiency video coding', ETRI Journal, vol. 36, no. 4, pp. 528-536.

[101] Wien, M 2015, 'High efficiency video coding', Coding Tools and Specification, pp. 133-160.

[102] Sector, IR 2015, 'Parameter values for the HDTV standards for production and International programme exchange', Recommendation ITU-R BT.709-5, p. 31.

[103] Tan, TK, Weerakkody, R, Mrak, M, Ramzan, N, Baroncini, V, Ohm, JR & Sullivan, GJ 2015, 'Video quality evaluation methodology and verification testing of HEVC compression performance', IEEE Transactions on Circuits and Systems for Video Technology, vol. 26, no. 1, pp. 76-90.

[104] Sze, V, Budagavi, M & Sullivan, GJ 2014, 'High Efficiency Video Coding (HEVC)', Integrated Circuit and Systems, Algorithms and Architectures, pp. 1-375.

[105] Huynh-Thu, Q & Ghanbari, M 2008, 'Scope of validity of PSNR in image/video quality assessment', Electronics Letters, vol. 44, no. 13, pp. 800-801.

[106] Jain, HB & Rao, KR 2014, 'Fast Intra- mode decision in high efficiency video coding', Polibits, vol. 50, pp. 5-12.

[107] Gan, Y, Zhao, X & Zhang, Q 2015, 'Fast algorithm with early termination CU split and mode decision', Inter-national Journal of Control and Automation, vol. 8, no. 2, pp. 83-94.

[108] Ma, Y, Liu, Z, Wang, X & Cao, S 2018, 'Fast Intra- coding based on CU size decision and direction mode decision for HEVC', Multimedia Tools and Applications, pp. 1-23.

[109] Karwowski, D & DomaĹ, M 2016, 'Context-adaptive binary arithmetic coding with precise probability estimation and complexity scalability for high-efficiency video coding', Journal of Electronic Imaging, vol. 25, no. 1, p. 013010.

[110] O'Neill, A, Hutter, M, Shao, W & Sunehag, P 2012, 'Adaptive context tree weighting', in Data Compression Conference, pp. 317-326.

[111] Lee, JH, Jang, KS, Kim, BG, Jeong, S & Choi, JS 2015, 'Fast video encoding algorithm for the Internet of things environment based on high efficiency video coding', International Journal of Distributed Sensor Networks, vol. 11, no. 11, p. 146067.

[112] Podder, PK, Paul, M & Murshed, M 2016, 'Fast mode decision in the HEVC video coding standard by exploiting region with dominated motion and saliency features', PloS One, vol. 11, no. 3, p. e0150673.

[113] Li, X, He, X, Peng, X & Xiong, S 2017, 'An image feature-based method to efficiently determine Inter-coding depth in HEVC', AEU-International Journal of Electronics and Communications, vol. 71, pp. 96-104.

[114] Purnachand, N, Alves, LN & Navarro, A 2012, 'Fast motion estimation algorithm for HEVC', in IEEE Second International Conference on Consumer Electronics-Berlin, pp. 34-37.

[115] Hsu, WJ & Hang, HM 2014, 'Fast coding unit decision algorithm for HEVC', in Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, pp. 1-5.

[116] Jiang, X, Wang, X, Song, T, Shi, W, Katayama, T, Shimamoto, T & Leu, JS 2018, 'An efficient complexity reduction algorithm for CU size decision in HEVC', International Journal of Innovative Computing, Information and Control, vol. 14, no. 1, pp. 309-322.

[117] Hore, A & Ziou, D 2010, ‘Image quality metrics: PSNR vs. SSIM’, in 20th IEEE Inter-national Conference on Pattern Recognition, pp. 2366-2369.

[118] Internet sources

List of Publications

1. Kanike Sreenivasulu, T V K Hanumatha Rao “A Neural Network-based Interframe Prediction for HEVC” International Journal of Engineering Trends and Technology (IJETT), ISSN: 2231-5381, Volume-70 issue-1,199-203, January,2022 (**SCOPUS**)
2. Kanike Sreenivasulu, T V K Hanumatha Rao “HEVC Intra Prediction through CNN Ensemble Learning” Neuroquantology,eISSN:1303-5150,Vol-20,Issue-6,Page7146-7153,48-53,June-2022 (**SCOPUS**)
3. Kanike Sreenivasulu, T V K Hanumatha Rao “Study and Comparison of H.265/HEVC Video Coding Architectures” International Journal For Advanced Researchs In Science & Technology (IJARST), ISSN:2320-1126, Vol-11, Issue-12, Page 48-53, Dec-2021
4. Kanike Sreenivasulu, T V K Hanumatha Rao “Efficient Image coding technique based on Seam Identification and Integer Wavelet Transform” International Journal for Innovative Engineering and Management Research(IJIEMR),ISSN: 2456-5083, Vol-11,Issue-3,Page 52-57, March, 2022.