End to End Video Compression Based on Deep-Learning

Hajar Yaseen and Siddeeq Ameen

November 14, 2021

End to End Video Compression Based on Deep-Learning

Abstract*— Recent years have shown exponential growth in video processing and transfer through the Internet and other applications. With the restriction on bandwidth, processing and storage there is an extensive demand for end-to-end video compression. Many conventional methods have been developed to compress video. However, with the extensive use of Artificial Intelligence, AI, such as Deep Learning (DL) have emerged as a best-of-breed alternative for performing different tasks have been also been used in the option of improving video compression in last years, with the primary objective of reducing compression ratio while preserving the same video quality. Evolving video compression research based on Neural Networks (NNs) focuses on two distinct directions: First; enhancing current video codecs by better predictions integrated even in the same codec framework, and second; holistic end-to-end VC systems approaches. Although some of the outcomes are optimistic and the results are well, no breakthrough has been reported previously. This paper review of new research work, including samples of few influential articles that demonstrate and further describe the various highlighted issues in the aria of using DL for end to end video compression.

***Keywords*—Deep Learning, Neural Networks, Convolutional Neural Networks, Video Compression, Intra-Prediction, Inter Prediction.**

I. Introduction

The outcrop of digital technology has led to the emersion of new ways to share information and knowledge on a global scale through the Internet. Most of the social networks rely mostly on image and video sharing, while online video streaming platforms have become very popular in recent years. It is predicted that in 2022 video streaming and downloads will exceed 82% of all consumer Internet traffic as expectations continue to grow [1].

Compared to pictures and other multimedia signals, video contains a large amount of information. For this reason, VC or encoding is an important tool that aims to reduce the size of digital video by taking advantage of the intrinsic redundancy [2]. These typically include spatial-temporal and information-based repetition. Recent developments in this area have resulted in compression rates as high as 1000:1, as is the case for the HEVC (High Efficiency Video Compression) video standard and further research and development is ongoing [3].

Traditionally, for VC, most algorithms use block-based predictive schemes along with residual compression. They make use of the advances in image coding to compress both key frames and the residuals between predicted and actual blocks [4]. However, this type of investigation is heavily guided by hand-crafted improvements and techniques. It is thus limited by the extent of the human understanding regarding the properties and statistical dependencies in both image and video [5].

There have been ups and downs in the popularity of NNs since they first appeared, as they compete with other AI methods for classification and prediction jobs. Over the last few years, the number of Deep Neural Networks (DNNs) applications has exploded, due largely to advancements in computer power and parallel processing via Graphics Processing Units (GPUs) [6]. As a best-of-breed option to replace traditional analytics algorithms in a number of applications, including identification, recognition, and classification, DL has risen to prominence in recent years [7]. DNN architectures appear to be a logical choice for VC algorithms that largely rely on predictions and filtering, given their performance and applications. Researchers are working on two main areas: learning-based optimization modules integrated with existing video/image codecs, and a purely learning compression framework [8]. There are several types of neural networks, each of which is well-suited to a specific categorization, recognition, or generating task. Fully Connected (FC), Convolutional Neural Networks (CNNs), and Long Short Term Memory (LSTM) Neural Networks are the most commonly utilized NNs variations for VC. Regression tasks are best handled by FC networks, whereas pattern detection is handled by CNNs, and learning from historical time series is handled best by LSTM [9]. VC uses these networks a lot because of their alignment with these characteristics.

In the remaining sections, we explore a variety of VC and DL ideas. Firstly, we provide a brief introduction to VC and evaluation metrics. Next, the DL with its types are illustrated. Then, we show how the various deep neural network layers that are key to deep compression systems work, especially in intra-prediction, inter-prediction and deep end-to-end VC framework. Finally, the adopted dataset and our assessment are discussed.

II. Video Compression

A video is a sequence of consecutive images acquired by projecting a real-world scene in a 2-D plane using a video capturing sensor or by creating a sequence of artificially generated images (animation). Each individual image, known as a frame or picture, is displayed with a certain frequency defined by the frame rate, generally expressed in frames per second (fps) or hertz (Hz). Frame rates can range from 24 fps to 30 fps, depending on the application, with 24 frames per second being the most common frame rate used in the film industry [10].

VC systems are composed of two main parts, an encoder and a decoder. The former is responsible for generating the compressed stream of bits (bitstream) from the input raw video file. The ratio between the bitrate of the compressed bitstream and the raw video file is known as the compression ratio. As the reverse process, the decoder is responsible for receiving a compressed bitstream as input and generating a raw displayable video file. Given that the bitstreams generated by the encoder need to be interpreted by the decoder, which is usually located in another device, these two systems need to be exactly compatible. For instance, for a video streaming service, encoding is performed at the data servers, while decoding is executed at the receiving device, such might be a television, a personal computer or even a mobile phone [11].

When it comes to compression, either lossless or lossy compression can be applied. Redundancy removal in video or image data using lossless compression. A reconstruction method which allows for perfect reconstruction at the cost of compression ratios of only shallow depth is provided. Lossy compression is irreversible since codecs reverse the reconstruction process to an approximation of the input data. Lossy codec research strives to reduce the compromise between compression and quality [12].

III. Evaluation Metrics

Evaluation metrics are focused on answering specific questions and addressing certain goals. Common metrics used to measure image quality compare two images: input and output. The purpose of image quality measurements is to rate the image quality in a way that is comparable to human judgement. This means that the perception of the human visual system should be approximated as accurately as possible using image quality measurements [13].

Compression approaches for both images and videos, in particular, are focused on optimizing for the peak signal-to-noise ratio (PSNR). Mean Square Error (MSE) is used, and the result is expressed in decibels (dB), for the input image X, and the output image Y, the syntax of PNSR defined as [14]:

$$PNSR(X,Y) = 10\log_{10}\left(\frac{M^2}{MSE}\right) \dots\dots\dots\dots\dots\,.\quad 1$$

Where M is the maximum value (Pixel) in the original image. The image pixel-wise statistical attributes are being compared only using this metric [14]. However, as previously discussed, PSNR has been an effective compression tool in the previous decades, but this technology lacks sufficient evidence that it outperforms SSIM (Structural SIMilarity) in regards to finding specific coding artifacts and other distortions, especially when compared to PSNR [15].

The SSIM is a more complicated metric that incorporates convolutional methods that apply a search window across an image and attempt to find a quality index of the image that isn't only calculated from pixel-by-pixel measurements but uses a broader receptive field to achieve better results. A general improvement is achieved with SSIM, but recently developed Multi Scale Structural Similarity (MS-SSIM) improves upon it by taking use of several applications of the SSIM metric at progressively lower scales of the image [16].

IV. Deep Learning Overview

DL is a type of machine learning (ML) that use numerous layers of increasingly complex algorithms to gradually reveal more detailed information from the raw input. In ML, a computational method is studied in order to understand its data-driven functionality, allowing it to learn how to do certain tasks using previous knowledge. Many applications, for instance, recommendation systems, search engines, digital assistants, and digital photography, employ these principles. The field of ML has evolved to incorporate other disciplines with quick technological breakthroughs and practical applications in the real world. Although ML algorithms are widely employed today, in the past the algorithms needed domain knowledge and specific features to help them interpret raw data. NNs have had a resurgence in recent years due to access to far more powerful computer equipment and massive datasets [17].

DL, or multi-layered neural networks, tends to have a more profound impact than shallower methods, hence it is referred to as DL. ML techniques have been mostly overcome by DL algorithms in nearly all computer vision applications. Additionally, these technologies are even able to surpass human participants in activities like as visual recognition or strategic games. Rather than developing a separate algorithm for each task, DL uses techniques that may be employed in a wide range of scenarios [18]. DNNs that have multiple layers or deep representations are said to have a “deep” or “profound” meaning. NNs excel in learning complex models with a large number of hidden variables and relationships, even with noisy data. Because of this, considerable study has been done on employing DL in both compression tasks, especially in image and VC [19].

DL takes a more holistic approach, studying the process of designing DNN structures as well as analyzing their performance. NNs and the optimization process will be discussed in detail in the next sections, followed by an overview of common DL approaches, which will highlight different ways for image and VC [20].

Most of the actions presented can be interchangeably used to produce alternative designs, however when developing networks, common design choices are demonstrated to deliver effective results. The typical cases are usually divided into classes of networks which share common characteristics, to facilitate the identification of networks [21]. Following, we present the three most popular types of DNNs used today.

A. Convolutional Neural Network

Computer vision, which is the field of computer systems design to recognize and learn from visual representations such as image, videos or other forms of multi-dimensional one specific form of DL model, Convolutional Neural Network (CNN) has been increasingly accepted by the community of computer vision [22]. CNN consist of a number of stacked convolutional layer and pooling layers, optionally. There are several trainable filters or kernels of a defined size (e.g., 3 x 3 or 5 x 5) in each convolutional layer that are successively applied to the outputs of previous layers. On the other hand, pooling layers combine the outcomes of these convolutions locally within nearby regions, decreasing the spatial dimensions of the representations and generating translational shift invariance. In addition, every convolution or pooling is applied to a block that is moved by a fixed number of positions, controlled by the step [23].

B. Recurrent Neural Network

A Recurring Neural Network (RNN) is named because neural network mathematics are repeated at every stage. This architecture takes account of the expected impact of the past on what happens in the future, which is why it is suitable for sequential data [24]. Neurons in the RNN have a "state" that can be understood as memory; they can recall important things that have happened and utilize this to predict next things. If your data are time series, the characteristics at ***t-4, t-3, ...*** and ***t-1*** may be taken to estimate what happens at ***t***. Trends and patterns previously witnessed are probably essential for anticipating what happens next [25].

C. Deep Auto Encoder

Autocoders are an unsupervised learning technique in which it use NNs to learn representation. The encoding is verified and enhanced by trying to regenerate the encoding input. The auto encoder learns how to represent a set of data, often to reduce dimensionality, by training the network to disregard inconsequential input [26].

Autoencoders always consist of an encoder unit and a decoder unit, which need to be simultaneously trained but can be utilized separately.

Autoencoders can be used to efficiently transform data into smaller spaces by ensuring the latent space is less than the original inputs. Due of their apparent parallels with a compression system, auto-encoders play a highly essential role in investigating various compression challenges via NNs [27].

V. Video Compression Using DL

The last decades have seen the emergence of a number of classic VC techniques, such as H.264 and H.265. This approach is used by predictive coding algorithms the vast majority of the time. They are manually designed, therefore they cannot be collaboratively optimized end-to-end. Intra prediction with residual coding, inter prediction as well as mode decision, entropy coding, and post-processing are some of the most often suggested DL-based approaches for VC. Rather than creating an end-to-end compression system, these techniques are utilized to upgrade a specific module of the standard VC algorithms. So, in this section we present the some of related research to intra-prediction, inter-prediction and end-to-end framework compression.

A. Intra-Prediction with DL

Intra-prediction is the most heavily researched field for enhancing VC methods using DNNs. I-Frames are images that contain the content of the pixels, which are compressed to save space. When partitioning the input video frame into Macro-Blocks, an intra-prediction encoding technique is applied, which reduces the need for I-frames, which generally have the highest bitrate. This is done by finding the previously scanned pixels in the next frame and correlating the data in each block. Once correlation is discovered, the following block pixels are predicted and only residual errors (differences) are supplied, resulting in better compression efficiency [28].

The latest research in the field has found that NNs perform very well at predicting future outcomes, which means in the last four years academics have been exploring the ways NNs can be used to do a better job of predicting the future. When it comes to classic prediction modes, the key benefit is the flexibility and adaptability of NNs due to the non-linear activation functions. Intra-prediction is only effective in the spatial domain, which is why it is possible to use it just as well for photos [29].

Classification CNN and supervised learning are used to assess blocks of images in [30] and train the network to determine the most likely best HEVC mode. These various modes, in particular the 33 angular Intra-Prediction modes, the DC mode, and the planar mode, are all considered possibilities. After completing the training process, the network is programmed. As can be seen in the block diagram (diagram, illustration), two convolutional, one max pooling, and two fully-connected layers are applied to each 32x32 block. Compared to a baseline of randomly selected modes, the RD-Loss values have been determined in this method.

In classic HEVC coding interstitial prediction approaches, ignoring the richer context between the current block and its surrounding blocks and so leading to inaccurate prediction is a concern, especially when there is a poor spatial link between the current block and the reference lines. To combat this challenge, W. Cui and et al. [31] suggested that an intra prediction neural network benefits from the rich context of the present block, thereby resulting in improved prediction accuracy. This network can be used to associate current block locations with reference block locations.

R. Birman and et al. [32], twelve linked NNs are employed to perform the prediction. MSE is reduced by three times when using computations that carry out three times as many operations as ordinary computation modes. The system investigated various trained network configurations to determine the original pixel values. One of these twelve network is built for forecasting one pixel at a time, while another one is built for forecasting four pixels. Adam optimizer was used to develop the Stochastic Gradient Decent (SGD) method, which is a refinement of the previous version of the network. The Authors have employed Python in order to build the network.

An idea of a novel block-wise prediction paradigm based on CNNs for lossless video coding is proposed by I. Schiopu and et al. [33], according to the analysis done by the authors. This is the first time that modern ML techniques have been used to replace all of the classic HEVC-based angular intra-prediction modes. Lossless HEVC Intra Prediction on the TUT-VTSEQ and HEVC-VTSEQ datasets shows a bit rate decrease of 5.8%.

C. Ma and et al. [34] are proposed method to perform the coefficient prediction to eliminate the coefficient redundancy. The trained CNNs are used to forecast the coefficients and apply it to HEVC intra-predicted residuals. To assist with coefficient prediction, an indication is provided to the decoder on whether to apply coefficient prediction or not at the coding unit level. While both the quantization and entropy coding phases precede the coefficient prediction step, the authors add this coefficient prediction step as an additional layer of complexity. Gradient descent algorithms are used to train the network. For the training data, UCID and DIV2K are employed. In the results, the technique achieved a mean BD-rate reduction ratio of 1.8% in Y, 4.1% in U, and 4.5% in V. Notably, the average drop in bit-rate (BD-rate) for 4K test sequences is 2.9%, 6.5%, and 6.6%.

B. Inter-Prediction Based On DL

Inter frame is a picture in a VC sequence that is described by referencing one or with reference to several other frames around it. The temporal redundancy between nearby frames in this prediction can lead to greater compression rates. A frame that is interceded is divided into macroblocks, or blocks. The encoder first encodes the raw pixel values for each block, and then searches for a previously encoded frame that is comparable to the one it is encoding [35]. The Inter-Prediction mechanisms that have been employed use two different types of prediction: bi-directional temporal prediction, which is based on the buffering of video frames and also future and past frame for the prediction, and forward temporal prediction, which uses a P-Frame to identify a matching block in the previous video frames (B-Frame). Further precision was also attained by incorporating the usage of matching block partial pixel displacements in the process [36].

The most popular current research focus was on utilizing CNNs to identify matching blocks characteristics and applying those to reduce the remaining prediction error of Inter-Prediction. When doing inter-prediction, CNNs are utilized to collect similarity of characteristics between consecutive frames [37].

Z. Zhao and et al., [38] present a technique to employ a CNN network that implements bi-directional motion correction weighting with improved accuracy. Training a CNN network to blend previous and future frames yields a projected frame that is more accurate than the one which uses a simple average of past and future frames. Approximately 10.5% BD-rate savings have been expected from the suggested approach and an average of 3.1% BD-rate savings compared to HEVC.

Another approach to improving Inter-Prediction accuracy was provided by J. K. Lee and et al., [39]. Full CNN Connected networks were trained to interpret motion compensated (Inter-Prediction) block pixel values from the previous frame as well as the next-neighboring block pixels. Utilizing the simplified motion compensated block, the network results have been better than those obtained using the temporal and spatial domain pixels combined into a single network input layer. Also, the author suggests using a virtual reference frame that uses video interpolation convolutional neural network for this. This frame correlates with the current frame more closely than any of the reference frames forward or back, allowing for smaller Motion Vectors and lower block residual values. The proposed solution has obtained an average of 1.4% HEVC BD-rate decrease.

Binary arithmetic coding, also referred as context-adaptive binary arithmetic coding, is utilized as the entropy coding method in HEVC. As they are unable to dynamically adapt to estimate the likelihood of syntax elements, the manually created binarization and context models should not be used. C. Ma and et al., [40] apply NNs to estimate the syntax elements' probabilities, and these probabilities are subsequently combined with the syntax element values to form an arithmetic coding engine. LDP-defined systems exclusively concern themselves with the syntactic parts of inter prediction information, such as merge flags, merge indexes, reference indexes, motion vector differences, and motion vector prediction indexes. This system has three new qualities. The first step is to bypass the surrounding syntax parts and directly feed them into the neural network. A second important feature of unified NNs is that they are better suited for prediction block sizes of varying sizes. In order to enhance parallelism, dependency among the syntactic elements has been removed from the current prediction unit. Stochastic gradient descent is used to train all the networks. The CDVL video data and SJTU training data are prepared using the video sequences. HM12.0 is very helpful for compressing video sequences since it produces training data at the decoder.

J. Lee et al. [41]. Propose a new video coding strategy that will use a CNN that reflects a convolutional neural network to enable improved motion prediction in (HEVC). They also designed a CNN and video prediction network (VPN), which uses a virtual private network to boost their ability to code effectively. Both end-to-end layer 2 VPNs are used to evaluate image at the same time. The virtual reference frame (VRF) is superior to the traditional reference frame because it provides more relevant details. The strategies described in this paper are used in the HEVC coding system. The VRF scheme succeeds in rate-distortion optimization among candidate sets using HEVC reference image without adding detail. The authors change the HEVC inter-prediction processes of AVS and MC prediction adaptively using PUIDFR as the reference point. This technique will leverage multi-hypothesis weighted prediction techniques in HEVC. It can be found in both RA and LD setups. High-definition (HD) and ultra-high-definition (UHD) videos are used for training, sourced from YouTube. As a result, compared to HEVC, 5.7% and 2.9% LD and RA coding improvement, respectively.

To improve bi-prediction accuracy in complex scenarios, H. Tao and et al., [42] has developed a novel inter prediction strategy that incorporates deep frame prediction network components. It should be noted that the suggested network uses multi-scale motion alignment, temporal and spatial correlation fusion, and frame synthesis modules. Through a network that can identify and precisely extract motion components, which can be manipulated across various scales, the system may utilize temporal and spatial correlation to produce a prediction frame that is driven by data. This helps us provide another prediction frame for bi-prediction because the system is included into VTM-6.2. Because of this, it is highly recommended that you include the prediction created by the proposed network in your reference list to broaden the range of references. According to this plan, on average, bi-prediction has allowed BD-rate savings of 1.8%

To enhance interpolation of reference samples required for fractional-precision motion correction, L. Murn and et al., [43] presents a unique neural network-based inter-prediction technique. A single neural network is required to be trained, after which a quarter-pixel interpolated filter set is created.

By using a training framework, each network branch may be thought of as matching a certain fractional shift. on average, lower resolution sequences under the, low-delay P, low-delay B and random-access configurations achieve an average of 2.25%, 1.27% and 0.77% BD-rate savings respectively, although the complexity of the learned interpolation schemes is significantly reduced compared to full CNNs.

C. End to End Compression Framework Based on DL

The end-to-end framework learning with non-linear transformation methods are not widely employed traditionally. These methods are intended to enforce DNN for video codec. But they only modify a few modules in the conventional framework instead of improving the system comprehensively [44]. There are two main problems in building standardized end-to-end video coding. Current learning-based video coding schemes cannot manipulate the advantages of end-to-end enhancement and are not conducive to multi-algorithm learning. Combining the benefits of traditional compressed data and neural network approaches is critical. It is necessary to find a scheme for motion detection or VC [45]. So, in this section we present some research work related to end-to-end compression framework that used DL.

G. Lu and et al., [46] taking the advantage of both traditional architectural methods in the classic model of VC and the solid non-linear capabilities of neural networks, they suggest the end-to-end model of video coding for the DNNs. Learning-based optical flow prediction is applied for obtaining the motion information. Next, the authors employ two auto encoders to encode the prediction motion information and remaining information, respectively. The whole modules are introduced to coordinate with others by considering the exchange between reducing the compressed bits and resolution of the decoded video.

A. Djelouah and et al., [47] presents an inter-frame compression approach to neural V that can smoothly build on different available neural image codecs. Their real end-to-end fix performs time prediction by optical flow-based motion compensation in image pixels. The primary point is that decoding efficiency and reconstruction quality can be enhanced by encoding the necessary information into a latent recognition that directly decodes into motion and blending coefficients. Residual information between the original image and the interpolated frame is needed to account for the remaining prediction errors. Propose to compute residues directly in latent space instead of in pixel space as this allows the same image compression network to be reused for both key frames and intermediate frames.

O. Rippel and et al., [48] propose a learned end-to-end video coding technique for low-latency systems, where there is just forward-looking information in each frame. They postulate two important points. The primary innovation of this design is a novel VC architecture that (1) generalizes motion estimation to deal with compensation learned by the model that is other than simply translation, (2) retains a state for additional information learned by the model instead of strictly relying on previously transmitted reference frames, and (3) merges the compression of all transmitted signals (such as optical flow and residual). Secondly, they provide a framework for using ML to do spatial rate control: a method for varying bitrates on a frame-by-frame basis. This is an important component for video coding, as they did not previously realize ML was possible with a computer.

M. Akin and et al., [49] offer a first-time end-to-end implementation of motion-compensated, hierarchical and bi-directional trained optical codec. The researchers developed centralized bi-directional flow prediction, flow compression, and frame estimation within a convolutional neural network system. All the system frameworks are constructed by learning filters and used by single DT- loss in the end.

Scale-space flow and scale-space warping, which are extensions of flow and bilinear warping, are introduced by E. Agustsson and et al., [50] as generalizations of these earlier concepts for application in motion correction in learnt VC. Bi-linear warping cannot adequately simulate regions that are slow or irregular in motion because of stripped off or a combination of both, as demonstrated by our study.

J. Pessoa and et al., [51] avoids explicit motion prediction and instead attempts to compresses videos by optimizing an auto-encoder architecture that is designed to capture both spatiotemporal structure and entropy. In addition to developing a 3D latent-space representation of video, they also implement a rate-distortion optimization structure that provides temporal consistency across frames, eliminating undesirable artifacts in the output video sequence. An end-to-end feature learning system using spatiotemporal auto-encoders for video use, where loss function optimizations account for reconstruction distortion and an entropy-coded quantized latent representation bit-rate (which includes an estimate of the length of the auto-encoder's latent representation).

N. Zou and et al., [52] have designed an end-to-end learning system for VC. Instead of basing their system on pixel-space motion, the system creates and stores latent representations of individual frames. Attention mechanisms are incorporated at the decoder to attend to the latent space of frames, which are combined to make the predicted current frame. By applying relevance masks that depend on the feature channels, spatial-varying channel allocation is achieved. The model is trained to minimize the bitrate by minimizing two losses: one for arithmetic coding and the other for context modeling.

F. Racapé and et al., [53] provides an end-to-end ANN-based compression architecture that uses bidirectional prediction. This codec handles video sequences broken into Groups of Pictures (GOPs) where each frame is being encoded in Intra or Inter mode. This allows for efficient hierarchical GOP temporal networks by leveraging previous and future decoded frames to anticipate inter frames. The authors investigate the benefits of improving the compression of motion information prediction residuals using specialized auto-encoder models with GOP-conditioned layers. The network is completely retrained.

VI. Adopted Datasets

To take advantage of current advances in AI technology, video coding algorithms are increasingly using DL methods such as CNNs, which have proven to deliver significant coding advantages over traditional approaches based on classic computer vision theory. Using these learning-based compression approaches, a lot more training material is required even than traditional compression or existing ML methods. These should have a wide range of material in a variety of formats and textures [54].

ML algorithms rely heavily on training datasets to get the best results. In order to ensure good model generalization and avoid potential over-fitting problems, a well-designed training dataset is required. There is no publicly available database exists that is specifically designed for learning video coding, as far as we are aware. The majority of the time, researchers have used training datasets created for other purposes (like super-resolution, frame interpolation, and classification) in their research until now [55]. The following list summarizes notable free image and video training datasets.

There are several notable free image and video training datasets available on the Internet for free, for example **ImageNet** [56], **DIV2K** [57], **Vimeo** [58], **REDS** [59], **MCML** [60], **SJTU** [61]) and others. These databases differ in terms of the number of image/videos and the percentage of resolution. Some of these databases was created primarily to aid in the recognition of visual objects (ImageNet), segmentation (DIV2K), and some for network training of video coding (Vimeo, REDS, MCML, SJTU).

High spatial resolutions and bit depths need modern video coding techniques that can handle information with a variety of texture types. The JVET Common Test Conditions (CTC) dataset, for example, contains conventional test sequences such as video clips with UHD quality (2160p) and 10 bit depth, as well as several static and dynamic texture variations.

VII. Assessment and Discussion

Predicting the most appropriate Intra-Prediction mode [30] is limited by the block's most accurate prediction mode, therefore there will be no enhancements over current standards by default. In order to further reduce error, residual errors are predicted. In our knowledge, trained networks, which reduce some residual errors, can also rice some of the errors. This means that generalizing the approach to huge datasets is a bit of a stretch. The authors of [31] were able to build a network that consistently improves the BD-Rate for various frame sequences, hence their strategy is considered practical. The longer-term spatial variation that is not taken into account when training the network makes it difficult to predict the full block pixel values from neighboring pixels. The authors' significant improvement [32] (through their use of twelve learning networks to triple the MSE compared to standard models) comes at the expense of more comprehensive computations. This means that increasing the complexity of the computational operations with the use of a large number of learning networks will increase the time taken for the video compression process. The proposed research by [33] outperformed all the research mentioned in this aspect by replacing the traditional HEVC prediction models with intelligent prediction models, as it outperformed HEVC by 5% in terms of improving the video coding system. Our approach suffers from the distribution of quantization errors so it won't do well for compression. There's the chance of exploring within the framework of lossless VC algorithms.

The proposed methods [38, 39, 40, 41, 42, 43] are differed for improving prediction with CNNs and are not interconnected. Each proposed method has its advantages and cannot be directly compared to any other way. The different proposed algorithms can be applied simultaneously and independently to take advantage of all the improvements. Improvements to block predictions and Motion Estimation (ME) accuracy are the primary goals, as this reduces the extent of the coded values and enables lower bit rates to be used. Training neural networks for more accurate and direct ME vectors predictions in this field (intelligent inter prediction) is expected to reduce residual error size between it and original MV, as well as providing an accurate search location for the latter, with the potential to reduce computation time during research. Most previously mention's researches (intelligent inter prediction) are configured with JCT-VC common test condition (CTC) [62] and used anchor in the simulation results. Depending on the researches results, we noted that the system [38] have a best result compare to another, which is save 3.1% BD-rate for random access (RA) compare to HEVC, which is use CNN to enhance the region that cannot be well handle with traditional bi-direction, such as the boundaries of a moving object. The system [42] also gain a good result (1.8%) by extract and fuse motion characteristics at multiple scales, and fully leverage temporal and spatial correlation in order to build the perdition frame. In Low-Delay (LD), the system [41] get a best result than other research, which is save 5.7% BD-rate compare to HVC, which investigate a new prediction network employing a previously coded frame, enhanced motion vector prediction, and merge prediction

The learning-based end-to-end framework [47, 48, 49, 50, 51, 52, 53] outperforms the conventional learning-based framework on structure similarity by avoiding block artifacts and processing the frame's full spatial information. They have achieved comparable or better performance than H.265/HEVC with the default setting on PSNR or MS-SSIM evaluation metric. Yet, none of the above works, can clearly beat traditional codecs when it comes to high bitrate compression, especially H.265 and H266. As the modern video coding systems like H265 and H266 have many computational requirements, therefore, learning-based end-to-end compression requires powerful hardware, like a GPU, to solve compression upon both encoder and decoder sides. Because earlier leverage required simpler network design, the issue has gotten worse. This means that researchers will have to incorporate the real-world application scenario into framework and network design so that an unbalanced framework and a lightweight network architecture can be thoroughly studied. Several studies have attempted to solve this issue, mostly by utilizing model compression techniques. All of the model constructions in this part have been investigated thoroughly and comprehensively justified by using a specific codec structure to pinpoint the rate-distortion efficiency/complexity trade-off. Table I. shows some of the differences in training dataset, testing dataset usage, experimental result and the advantage point for each deep compression model that mention in the section V/C.

TABLE I. Different Compares for End-to-End Based DL

Ref. Year	Training Dataset	Testing Dataset	Result	Advantage
[46] 2019	Vimeo-90k	UVG	PSNR≈38.8 dB for UVG MS-SSIM≈0.973 for UVG	Propose NNs are utilized throughout the whole VC process.
[47] 2019	Vimeo-90k	MCL-JVC, VTL and UVG	PSNR≈39.89 dB for MCL-JVC PSNR≈38.70 dB for VTL PSNR≈40.1 dB for UVG	Join image synthesis and motion compression in order to reduce the motion code size.
[48] 2019	HD Video from YouTube	CDVL SD and Xiph HD	For SD and HD, 60% and 20% greater than proposed model code produced. Respectively.	Any existing video codec can be outperformed by this model.
[49] 2020	REDS	REDS	PSNR≈36.6 dB exceed H264 and H265	Reduce the rate-distortion cost function averaged over image groupings
[50] 2020	Approx.	UVG MCL-JCV	Exceed HEVC up 0.05 bpp for MS-SSIM and up 0.15 for PSNR	Outperforms contemporary learning-based techniques and the mainstream codecs H.264 and HEVC when using MS-SSIMM.
[51] 2020	YouTube-8M	MCL-V	The model outperforms the baseline, compares better to MPEG-4 Part2 and H.265/HEVC at lower bitrates.	A single spatiotemporal auto-encoder (SAE) can serve as both reconstruction loss and entropy model train.
[52] 2020	CLIC	CLIC	MS-SSIM≈0.978 and PSNR≈30.44 dB at 0.0707 Bpp Decoder size=15.8MB decoding time=1484 seconds	Combines the current predicted frame with the previous frame to generate a more stable image.
[53] 2021	BVI-DVC	JVET (CTC)	On average PSNR≈35.3 dB	Construct a neural network codec and train it from the base up in order to upend the existing hybrid codec approach.

VIII. Conclusion

Lossy video coding seeks to represent the pixels of raw video as compactly as possible by leveraging the spatial-temporal and statistical connections. Traditional hybrid coding systems have employed all of the methods previously mentioned (such as pixel domain intra/inter prediction, transform, entropy coding, etc.) to serve this function over decades. Each coding tool is carefully studied under a specific codec architecture to explain the R-D efficiency against complexity tradeoff. This approach resulted in well-known industry standards including H.264/AVC, H.265/HEVC, and AV1. However, DNNs have shown powerful video spatiotemporal feature representation for vision applications including object segmentation and tracking. This presents the challenge of encoding spatio-temporal information in a compact manner for lossy compression. So, in this paper we present a summary of the principal methods of using DNN ability for video compression up to since.

IX. References

[1] J. W. Soh, J. Park, Y. Kim, B. Ahn, H.-S. Lee, Y.-S. Moon, *et al.*, "Reduction of video compression artifacts based on deep temporal networks," *IEEE Access*, vol. 6, pp. 63094-63106, 2018.

[2] C. Dong, Y. Deng, C. C. Loy, and X. Tang, "Compression artifacts reduction by a deep convolutional network," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 576-584.

[3] 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, pp. 1649-1668, 2012.

[4] T. S. Kumar, "A novel method for HDR video encoding, compression and quality evaluation," *Journal of Innovative Image Processing (JIIP),* vol. 1, pp. 71-80, 2019.

[5] S. Pandiammal, K. Rajalakshmi, and K. Mahesh, "An Enhanced Approach for Video Compression," 2018.

[6] T. Chen, H. Liu, Q. Shen, T. Yue, X. Cao, and Z. Ma, "DeepCoder: A deep neural network based video compression," in *2017 IEEE Visual Communications and Image Processing (VCIP)*, 2017, pp. 1-4.

[7] S. Mehta, C. Paunwala, and B. Vaidya, "CNN based Traffic Sign Classification using Adam Optimizer," in *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*, 2019, pp. 1293-1298.

[8] G. He, C. Wu, L. Li, J. Zhou, X. Wang, Y. Zheng, *et al.*, "A Video Compression Framework Using an Overfitted Restoration Neural Network," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2020, pp. 593-597.

[9] K. Yuan, B. Ying, S. Vlaski, and A. H. Sayed, "Stochastic gradient descent with finite samples sizes," in *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP)*, 2016, pp. 1-6.

[10] A. J. Brown and A. S. Baburin, "System and method for digital video management," ed: Google Patents, 2010.

[11] E. L. Ameres, J. Bankoski, A. W. Grange, T. Murphy, P. G. Wilkins, and Y. Xu, "Video compression and encoding method," ed: Google Patents, 2009.

[12] W. H. A. Bruls and R. B. M. K. Gunnewiek, "Video compression," ed: Google Patents, 2006.

[13] I. E. Richardson, *The H. 264 advanced video compression standard*: John Wiley & Sons, 2011.

[14] G. T. a. S. M. O. M. a. S. J. H. a. D. V. a. D. M. a. S. B. a. M. C. a. R. Sukthankar, "Variable Rate Image Compression with Recurrent Neural Networks," *arXiv preprint arXiv:1511.06085.*, 2016.

[15] A. Horé and D. Ziou, "Image Quality Metrics: PSNR vs. SSIM," in *2010 20th International Conference on Pattern Recognition*, 2010, pp. 2366-2369.

[16] D. R. I. M. Setiadi, "PSNR vs SSIM: imperceptibility quality assessment for image steganography," *Multimedia Tools and Applications,* vol. 80, pp. 8423-8444, 2021/03/01 2021.

[17] H. M. Yasin and A. M. Abdulazeez, "Image Compression Based on Deep Learning: A Review," *Asian Journal of Research in Computer Science,* pp. 62-76, 2021.

[18] G. V. M. Lakshmi, "Implementation of image compression using Fractal Image Compression and neural networks for MRI images," in *2016 International Conference on Information Science (ICIS)*, 2016, pp. 60-64.

[19] S. Zhu, C. Liu, and Z. Xu, "High-Definition Video Compression System Based on Perception Guidance of Salient Information of a Convolutional Neural Network and HEVC Compression Domain," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 30, pp. 1946-1959, 2020.

[20] H. Zhao, M. He, G. Teng, X. Shang, G. Wang, and Y. Feng, "A CNN-Based Post-Processing Algorithm for Video Coding Efficiency Improvement," *IEEE Access,* vol. 8, pp. 920-929, 2020.

[21] N. Aloysius and M. Geetha, "A review on deep convolutional neural networks," in *2017 International Conference on Communication and Signal Processing (ICCSP)*, 2017, pp. 0588-0592.

[22] C. Ma, D. Liu, X. Peng, L. Li, and F. Wu, "Convolutional Neural Network-Based Arithmetic Coding for HEVC Intra-Predicted Residues," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 30, pp. 1901-1916, 2020.

[23] A. Kamilaris, and Francesc X. Prenafeta-Boldú, "A review of the use of convolutional neural networks in agriculture.," *The Journal of Agricultural Science 156,* vol. 3, pp. 312-322, 2018.

[24] Y. Hu, W. Yang, S. Xia, W. Cheng, and J. Liu, "Enhanced Intra Prediction with Recurrent Neural Network in Video Coding," in *2018 Data Compression Conference*, 2018, pp. 413-413.

[25] Y. Yu, X. Si, C. Hu, and J. Zhang, "A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures," *Neural Computation*, vol. 31, pp. 1235-1270, 2019.

[26] A. Habibian, Ties van Rozendaal, Jakub M. Tomczak, and Taco S. Cohen, "Video compression with rate-distortion autoencoders," *In Proceedings of the IEEE/CVF International Conference on Computer Vision,* pp. 7033-7042, 2019.

[27] R. Yang, F. Mentzer, L. V. Gool, and R. Timofte, "Learning for Video Compression With Recurrent Auto-Encoder and Recurrent Probability Model," *IEEE Journal of Selected Topics in Signal Processing,* vol. 15, pp. 388-401, 2021.

[28] B. Heidari and M. Ramezanpour, "Reduction of intra-coding time for HEVC based on temporary direction map," *Journal of Real-Time Image Processing,* vol. 17, pp. 567-579, 2020.

[29] O. Hadar, A. Shleifer, D. Mukherjee, U. Joshi, I. Mazar, M. Yuzvinsky, *et al.*, "Novel modes and adaptive block scanning order for intra prediction in AV1," in *Applications of Digital Image Processing XL*, 2017, p. 103960G.

[30] T. Laude and J. Ostermann, "Deep learning-based intra prediction mode decision for HEVC," in *2016 Picture Coding Symposium (PCS)*, 2016, pp. 1-5.

[31] W. Cui, T. Zhang, S. Zhang, F. Jiang, W. Zuo, and D. Zhao, "Convolutional neural networks based intra prediction for HEVC," *arXiv preprint arXiv:1808.05734,* 2018.

[32] R. Birman, Y. Segal, A. David-Malka, and O. Hadar, "Intra prediction with deep learning," in *Applications of Digital Image Processing XLI*, 2018, p. 1075214.

[33] I. Schiopu, H. Huang, and A. Munteanu, "CNN-based intra-prediction for lossless HEVC," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 30, pp. 1816-1828, 2019.

[34] C. Ma, D. Liu, L. Li, Y. Wang, and F. Wu, "Convolutional Neural Network-Based Coefficients Prediction for HEVC Intra-Predicted Residues," in *2020 Data Compression Conference (DCC)*, 2020, pp. 183-192.

[35] H. Li, Z. Li, and C. Wen, "Fast mode decision algorithm for inter-frame coding in fully scalable video coding," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 16, pp. 889-895, 2006.

[36] Y. Suzuki, C. S. Boon, and T. K. Tan, "Inter frame coding with template matching averaging," in *2007 IEEE International Conference on Image Processing*, 2007, pp. III-409-III-412.

[37] D. Ding, L. Kong, W. Wang, and F. Zhu, "A progressive CNN in-loop filtering approach for inter frame coding," *Signal Processing: Image Communication*, vol. 94, p. 116201, 2021.

[38] Z. Zhao, S. Wang, S. Wang, X. Zhang, S. Ma, and J. Yang, "CNN-based bi-directional motion compensation for high efficiency video coding," in *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2018, pp. 1-4.

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

[40] C. Ma, D. Liu, X. Peng, Z. Zha, and F. Wu, "Neural Network-Based Arithmetic Coding for Inter Prediction Information in HEVC," in *2019 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2019, pp. 1-5.

[41] J.-K. Lee, N. Kim, S. Cho, and J.-W. Kang, "Deep video prediction network-based inter-frame coding in HEVC," *IEEE Access,* vol. 8, pp. 95906-95917, 2020.

[42] H. Tao, J. Qian, L. Yu, and H. Wang, "Bi-Prediction Enhancement with Deep Frame Prediction Network for Versatile Video Coding," in *2021 Data Compression Conference (DCC)*, 2021, pp. 374-374.

[43] L. Murn, S. Blasi, A. F. Smeaton, and M. Mrak, "Improved CNN-based Learning of Interpolation Filters for Low-Complexity Inter Prediction in Video Coding," *arXiv preprint arXiv:2106.08936,* 2021.

[44] Z. Zhao, S. Wang, S. Wang, X. Zhang, S. Ma, and J. Yang, "Enhanced bi-prediction with convolutional neural network for high-efficiency video coding," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 29, pp. 3291-3301, 2018.

[45] B. S. Kumar and V. U. Shree, "AN END-TO-END VIDEO COMPRESSION USING DEEP NEURAL NETOWRK."

[46] G. Lu, W. Ouyang, D. Xu, X. Zhang, C. Cai, and Z. Gao, "DVC: An End-To-End Deep Video Compression Framework," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 10998-11007.

[47] A. Djelouah, J. Campos, S. Schaub-Meyer, and C. Schroers, "Neural Inter-Frame Compression for Video Coding," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 6420-6428.

[48] O. Rippel, S. Nair, C. Lew, S. Branson, A. G. Anderson, and L. Bourdev, "Learned video compression," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 3454-3463.

[49] M. Akin Yilmaz and A. Murat Tekalp, "End-to-End Rate-Distortion Optimization for Bi-Directional Learned Video Compression," *arXiv e-prints,* p. arXiv: 2008.05028, 2020.

[50] E. Agustsson, D. Minnen, N. Johnston, J. Balle, S. J. Hwang, and G. Toderici, "Scale-space flow for end-to-end optimized video compression," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8503-8512.

[51] J. Pessoa, H. Aidos, P. Tomás, and M. A. Figueiredo, "End-to-end learning of video compression using spatio-temporal autoencoders," in *2020 IEEE Workshop on Signal Processing Systems (SiPS)*, 2020, pp. 1-6.

[52] N. Zou, H. Zhang, F. Cricri, H. R. Tavakoli, J. Lainema, E. Aksu, *et al.*, "End-to-end learning for video frame compression with self-attention," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020, pp. 142-143.

[53] F. Racapé, J. Bégaint, S. Feltman, and A. Pushparaja, "Bi-directional prediction for end-to-end optimized video compression," in *Applications of Digital Image Processing XLIV,* 2021, p. 1184205.

[54] S. Ma, X. Zhang, C. Jia, Z. Zhao, S. Wang, and S. Wang, "Image and Video Compression With Neural Networks: A Review," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 30, pp. 1683-1698, 2020.

[55] D. Liu, Y. Li, J. Lin, H. Li, and F. Wu, "Deep learning-based video coding: A review and a case study," *ACM Computing Surveys (CSUR),* vol. 53, pp. 1-35, 2020.

[56] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision,* vol. 115, pp. 211-252, 2015.

[57] E. Agustsson and R. Timofte, "Ntire 2017 challenge on single image super-resolution: Dataset and study," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2017, pp. 126-135.

[58] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, 2001, pp. 416-423.

[59] S. Nah, S. Baik, S. Hong, G. Moon, S. Son, R. Timofte, *et al.*, "Ntire 2019 challenge on video deblurring and super-resolution: Dataset and study," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 0-0.

[60] M. Cheon and J.-S. Lee, "Subjective and objective quality assessment of compressed 4K UHD videos for immersive experience," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 28, pp. 1467-1480, 2017.

[61] L. Song, X. Tang, W. Zhang, X. Yang, and P. Xia, "The SJTU 4K video sequence dataset," in *2013 Fifth International Workshop on Quality of Multimedia Experience (QoMEX)*, 2013, pp. 34-35.

[62] F. Bossen, "Common test conditions and software reference configurations," *JCTVC-L1100, 12th JCT-VC meeting, Geneva,* 2013.