Semantic Communication Based Video Coding Using Temporal Prediction of Deep Neural Network Parameters

Prabhath Samarathunga
Department of Computer and Information Sciences
University of Strathclyde
Glasgow, UK
prabhath.samarathunga@strath.ac.uk

Yasith Ganearachchi
Department of Computer and Information Sciences
University of Strathclyde
Glasgow, UK
yasith.ganearachchi@strath.ac.uk

Thanuj Fernando
Department of Computer and Information Sciences
University of Strathclyde
Glasgow, UK
thanuj.fernando.2023@uni.strath.ac.uk

Indika Alahapperuma
Department of Computer and Information Sciences
University of Strathclyde
Glasgow, UK
indika.alahapperuma@strath.ac.uk

Anil Fernando
Department of Computer and Information Sciences
University of Strathclyde
Glasgow, UK
anil.fernando@strath.ac.uk

***Abstract*—Video coding is a critical capability that underpins gaming, entertainment and media ecosystems, enabling effective use of video content in both conventional and non-conventional formats. Semantic communications, where semantics alone can be used to reconstruct media content provided that the context of semantic extraction is known, can effectively implement video coding, but techniques to exploit temporal correlations between video frames to achieve better rate distortion performance with them are just beginning to evolve. A novel approach for this problem of predicting the semantic decoder parameters using temporal correlation is proposed and tested using an autoencoder-based semantic communication system, and the performance is compared with the Neural Network Encoder-Decoder (NNCodec). Experimental results show that it achieves significantly better rate distortion performance compared to NNCodec alone, with PSNR gains between 3 and 25 dB depending on the complexity of the video and an average bitrate saving of 54%.**

***Index Terms*—Autoencoders, Deep Neural Networks, NNCodec, Semantic Communications, Video Transmission**

I. Introduction

Video has now become an integral component of the gaming, entertainment and media ecosystems and is available in a wide range of formats which include conventional 2D videos as well as a wide range of non-conventional formats, such as computer generated imagery (CGI), screen content video and 360° video [1], [2]. This widespread adaptation of video, which is consumed in an even wider variety of devices, has only been possible due to advancements in video compression by exploiting the inherent statistical and perceptual redundancies in video to reduce their huge file sizes to manageable sizes so that they can be efficiently stored, transmitted, and viewed within the constraints imposed by device and network capabilities [3].

We propose a novel semantic communication based video coding system using temporal prediction of deep neural network (DNN) parameters to exploit the spatial and temporal correlations in video, and test it using an autoencoder-based implementation where we attempt to predict the receiver-side decoder DNN parameters using key frames and displacement vectors of the parameters. Although prediction of DNN parameters in the spatial domain (within a given topology) has been explored [4], [5] and is used in NNC techniques to some extent, this is a first attempt to exploit the inherent spatial and temporal correlations in video to predict temporal relationships in DNN parameters of a semantic communication based video coding system.

The novel contributions from this work for semantic communication based video coding are:

- Use of DNN based semantic communication system for video coding and transmission
- Introducing a technique for temporal prediction of DNN parameter in video applications
- Demonstrating the effective combination of DNN and NNC for video coding and transmission

II. Related Work

Video coding standards are used to standardize video compression systems for wide interoperability and are conventionally based on digital signal processing (DSP). These systems operate by exploiting the statistical and perceptual redundancies of video in both spatial and temporal domains using

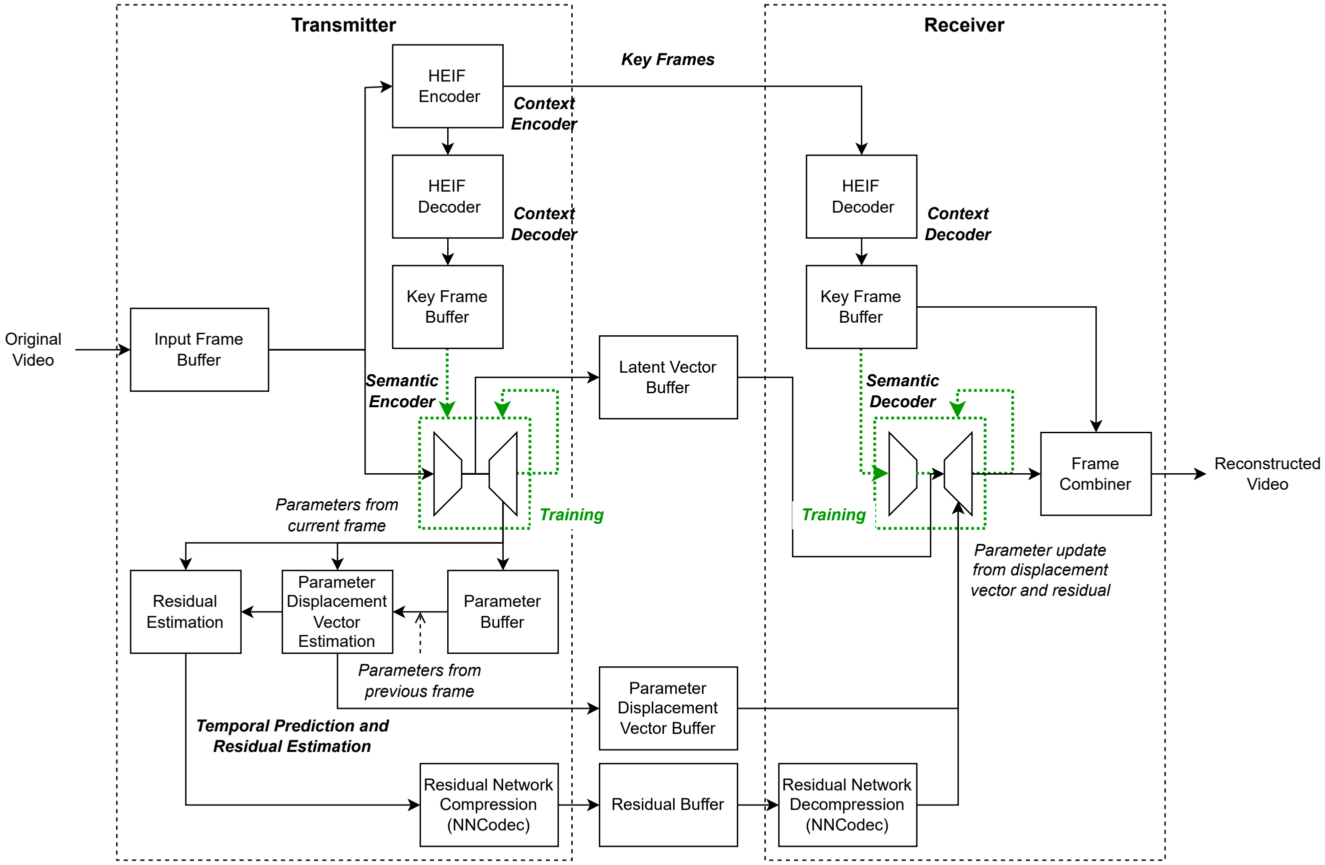

Transmitter
Receiver
Key Frames
HEIF Encoder
Context Encoder
HEIF Decoder
Context Decoder
Key Frame Buffer
Original Video
Input Frame Buffer
Latent Vector Buffer
Semantic Encoder
Semantic Decoder
Training
Frame Combiner
Reconstructed Video
Parameters from current frame
Residual Estimation
Parameter Displacement Vector Estimation
Parameter Buffer
Parameter update from displacement vector and residual
Parameters from previous frame
Parameter Displacement Vector Buffer
Temporal Prediction and Residual Estimation
Residual Network Compression (NNCodec)
Residual Buffer
Residual Network Decompression (NNCodec)

Fig. 1. Proposed system for video coding using temporal prediction of deep neural network parameters

advanced digital signal processing techniques, with state-of-the-art codecs such as Versatile Video Coding (VVC) implementing advanced motion prediction and compensation techniques which have evolved for over four decades. However, the exponential growth of video content, especially in non-conventional formats, along with the increase in resolution, frame rate, and color depth, is becoming a challenge even for state-of-the-art video coding systems [6]. In response, parallel to improving the performance of conventional DSP-based video coding standards, exploration of non-linear transform-based video coding systems using DNN is being carried out [7], although no widely accepted system has yet emerged.

Provided the inherent spatial and temporal redundancies contained in video frames, especially within a scene, an alternative approach is to train the encoder and decoder at the transmitter on a Group of Pictures (GoP) basis, which will intentionally overfit DNN parameters to a given scene, minimizing the reconstruction loss, and then update the decoder in the receiver for each GoP. This approach also requires a method to compress and transmit the decoder layers to the receiver, which diminishes any gain achieved through the DNN or semantic communication based compression even when using an NNC technique, as a significant overhead will need to be transferred in the form of the compressed DNN layers for each GoP or scene. An alternative is to make use of key frames of a GoP, which form the *context* when implementing a semantic communication based video coding system, to remotely train the receiver-side decoder which will be initialized using the same initializer and seed as a similar decoder implemented in the transmitter-side solely for the purpose of remote training. When augmented by a suitable residual coding system, this setup should theoretically be able to perform on par with state-of-the-art video coding systems but requires creating additional system complexity.

Semantic communications [8], which operate on the concept that only the *semantics* of a message is sufficient to reconstruct the original in a receiver provided the *context* used to extract the *semantic* in shared between the transmitter and receiver, has received recent attention as a method to implement effective image [9], [10] and video [11], [12] compression systems. Practical implementation of semantic communications is only possible due to advances made in artificial intelligence and machine learning (AI/ML), and DNN architectures, such as autoencoders (AE), convolutional neural networks (CNN), and transformers, have been widely used in developing such systems.

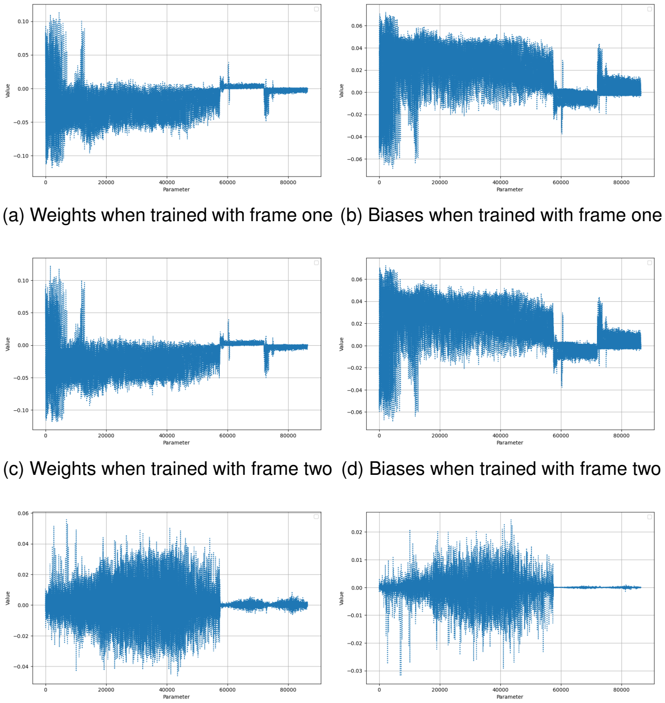
A key challenge in implementing DNN and semantic communication based video coding systems is that the decoder has to be trained along with the encoder at the transmitter, which then has to be taken to the receiver to decode the compressed bit stream. A common approach is to initially train the DNN with a large training data set so that the decoder is compatible with any video and then use a neural network compression (NNC) [13] technique to store and transport it to the receiver, but generalization of the DNN parameters results in low rate distortion performance compared to the original video. This can provide satisfactory performance for applications which are not meant for human perception, but are not be able to achieve equivalent rate distortion performance compared with state-of-the-art DSP based video coding standards. In scenarios where trained DNN models must traverse the network, the need for a larger bandwidth emerges as a major constraint. This challenge escalates when network transmissions become frequent due to changing conditions, which will lead to a change in the DNN model. To address this issue, both academia and industry have proposed the concept of NNC [13]. This approach involves the utilization of various techniques to compress neural networks, which are then transmitted over network channels [14]–[16]. On the receiving end, the compressed neural network is decompressed, making it ready for use in inference tasks.

An innovative approach to improve the fidelity of the reconstructed video from a semantic communication based video can be sought by seeking inspiration from DSP based video coding, where the temporal redundancies between frames are identified and used to predict the next frame at the decoder. However, instead of attempting to predict the motion of pixel groups based on temporal correlation, a DNN such as an autoencoder, can be trained at the transmitter using each frame of the video, and the changes of the DNN parameters between frames can be extracted and used at the receiver to predict the parameters that can produce an excellent estimation of the parameters for the decoder layers. If the transmitter-side encoder is trained to intentionally overfit each frame and reconstruct it with minimal reconstruction loss, transmitting the latent vector from the bottleneck and displacement vector of the DNN parameters are sufficient to reconstruct a very high-fidelity video at the receiver with much less complexity than the approaches discussed up to now.

III. Proposed System

The proposed system comprises four main functional components: context encoder/decoder pair, semantic encoder/ decoder pair, temporal prediction and residual estimation network, and a residual network compression/decompression system, as shown in Fig. 1. The video is converted using 4:2:0 chroma subsampling (YUV420) before processing and is read into an input frame buffer.

The context encoder/decoder pair provides the *context* used to extract the *semantics* at the transmitter and reconstruct the video from the *semantics* at the receiver. This is implemented by extracting the first frame, or key frame, of each scene of the video, with which subsequent frames have a strong temporal correlation. The key frame, which still contains spatial redundancies, is compressed using High Efficiency Image Format (HEIF).



(a) Weights when trained with frame one (b) Biases when trained with frame one

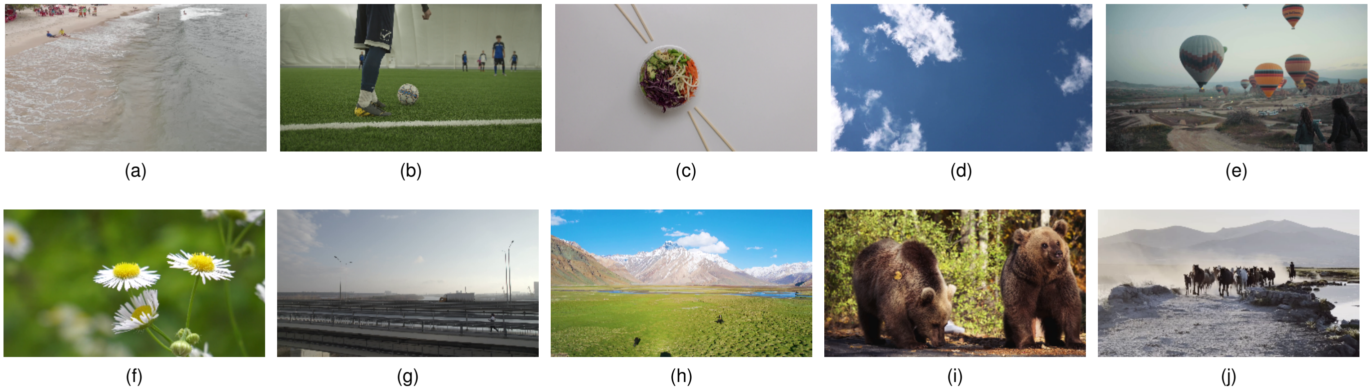
(c) Weights when trained with frame two (d) Biases when trained with frame two

(e) Difference of weight between frames (f) Difference of bias between frames

Fig. 2. Distribution of values of parameters (weights and biases) of the decoder layer when trained with two consecutive frames of a video. Note: the vertical axes of (e) and (f) are of higher resolution.

The semantic encoder and decoder are implemented using an autoencoder, which is trained over multiple epochs with the frames from the video to intentionally overfit it to a given frame. The autoencoder is initialized using the Glorot uniform initializer [27] and a fixed seed value which is shared with the decoder. To enable temporal prediction, the autoencoder is first trained using the key frame (using a *context* decoded version to match it with the receiver) until the peak signal-to-noise ratio (PSNR) between the input and output reach infinity such that they are identical. Then the latent vector of the autoencoder is read to a latent vector buffer and transmitted to the receiver, and the parameters (weights and biases) of the decoder network are read to a parameter buffer. The next frame of the scene is trained starting with the same parameters until the PSNR reaches infinity and the latent vector is extracted. This process is repeated for each frame in the scene and is restarted when a new scene is identified using an appropriate scene transition detection algorithm [28].

At the receiver, an identical autoencoder (initialized with the Glorot uniform initializer using the same seed value as on the transmitter side) is trained using the received key frame until the PSNR reaches infinity. For subsequent frames, the parameters of the decoder layers are updated using the displacement vectors received from the temporal parameter prediction and residual estimation network and the quantized residual received from the residual network compression/ decompression system. The updated decoder layers are used


(a) (b) (c) (d) (e)

(f) (g) (h) (i) (j)

Fig. 3. Video Clips used for Experiments: (a) Video 1 [17], (b) Video 2 [18], (c) Video 3 [19], (d) Video 4 [20], (e) Video 5 [21], (f) Video 6 [22], (g) Video 7 [23], (h) Video 8 [24], (i) Video 9 [25], (j) Video 10 [26].

to predict the next frame based on the received latent vector, and the reconstructed frame is collected in a frame combiner buffer, where the video is reconstructed.

The major innovation proposed is the temporal prediction and residual estimation used to update the parameters of the semantic decoder in the receiver. This exploits the temporal correlation between each frame of a scene in a video, which corresponds to a temporal correlation between each frame of a video. An example is shown in Fig. 2, which shows that the weights and biases of subsequent frames within a scene are nearly identical, with very small differences between the two.

Furthermore, since motion between two frames only occurs in specific parts of the frame rather than the frames changing altogether, the parameter values corresponding to pixel locations of the second frame can be observed to have moved in its location within the parameter vector when compared with the first frame. This can be used to calculate a parameter displacement vector (v_k) for the parameters corresponding to a predefined block of pixels in the frame by searching the parameters of the previous frame so that the mean squared error (MSE) between the predicted and actual blocks is minimized, as shown in (1) where p is the block size, n is the current frame, k is the block number.

$$MSE_k = \frac{1}{p}\sum_{i=1}^{p}(w_n(t_{k+i}) - w_{n-1}(t_{k+i}))^2 \tag{1}$$

v_k is then derived by minimizing MSE_k within a range defined by j within an arbitrary limit $\pm a$, as shown in (2).

$$v_k = \min(MSE_{k.j}),\ j \in (-a : a) \tag{2}$$

In the proposed system, the parameter displacement vectors are calculated by comparing the current parameters of the decoder layers with the previous parameters of the decoder layers of the parameter buffer. These are then written into a parameter displacement vector buffer and sent to the receiver to estimate the decoder parameters of each frame.

Parameter displacement vectors are used to estimate the decoder parameters in the transmitter, and a residual is extracted by comparing it with the current parameters. This residual, which is a vector of neural network parameters, is compressed using NNCodec to reduce the size of the residual and is then added to the update of the decoder parameters at the receiver. The level of quantization in NNCodec [29] can be varied using a quantization parameter (QP) to change the quality of the reconstructed video.

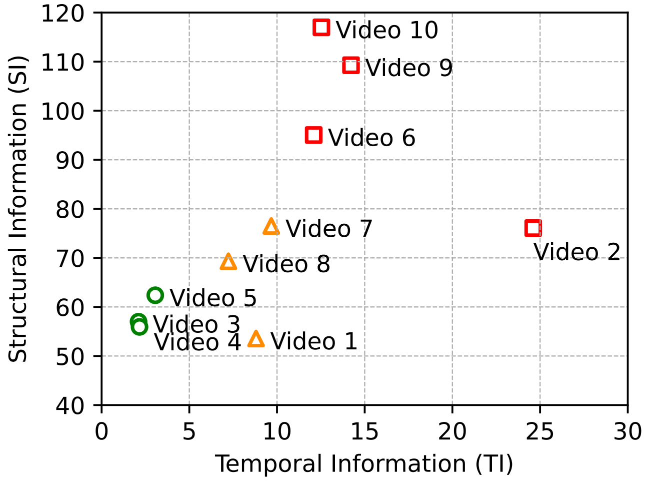
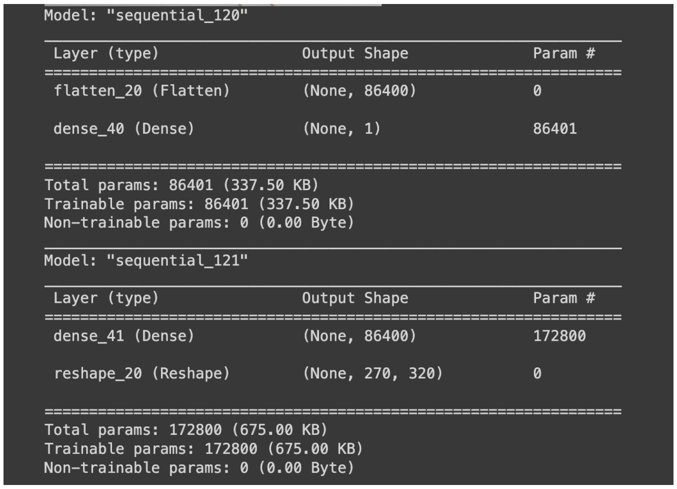


Fig. 4. Structural Information (SI) and Temporal Information (TI) in Test Videos

IV. Results and Discussion

The proposed system is tested using ten videos (Fig. 3) of spatial resolution 320×180 with varying structural information (SI) and temporal information (TI) content.

Fig. 4 shows the TI and SI content of each video, where videos 3, 4 and 5 are considered as low complexity videos (low TI and low SI), videos 1, 7, and 8 are considered as medium complexity videos (medium TI and medium SI), and videos 2, 6, 9 and 10 are considered as high complexity videos (high TI and/or high SI).



```
Model: "sequential_120"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 flatten_20 (Flatten)        (None, 86400)             0

 dense_40 (Dense)            (None, 1)                 86401

=================================================================
Total params: 86401 (337.50 KB)
Trainable params: 86401 (337.50 KB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
Model: "sequential_121"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_41 (Dense)            (None, 86400)             172800

 reshape_20 (Reshape)        (None, 270, 320)          0

=================================================================
Total params: 172800 (675.00 KB)
Trainable params: 172800 (675.00 KB)
Non-trainable params: 0 (0.00 Byte)
```

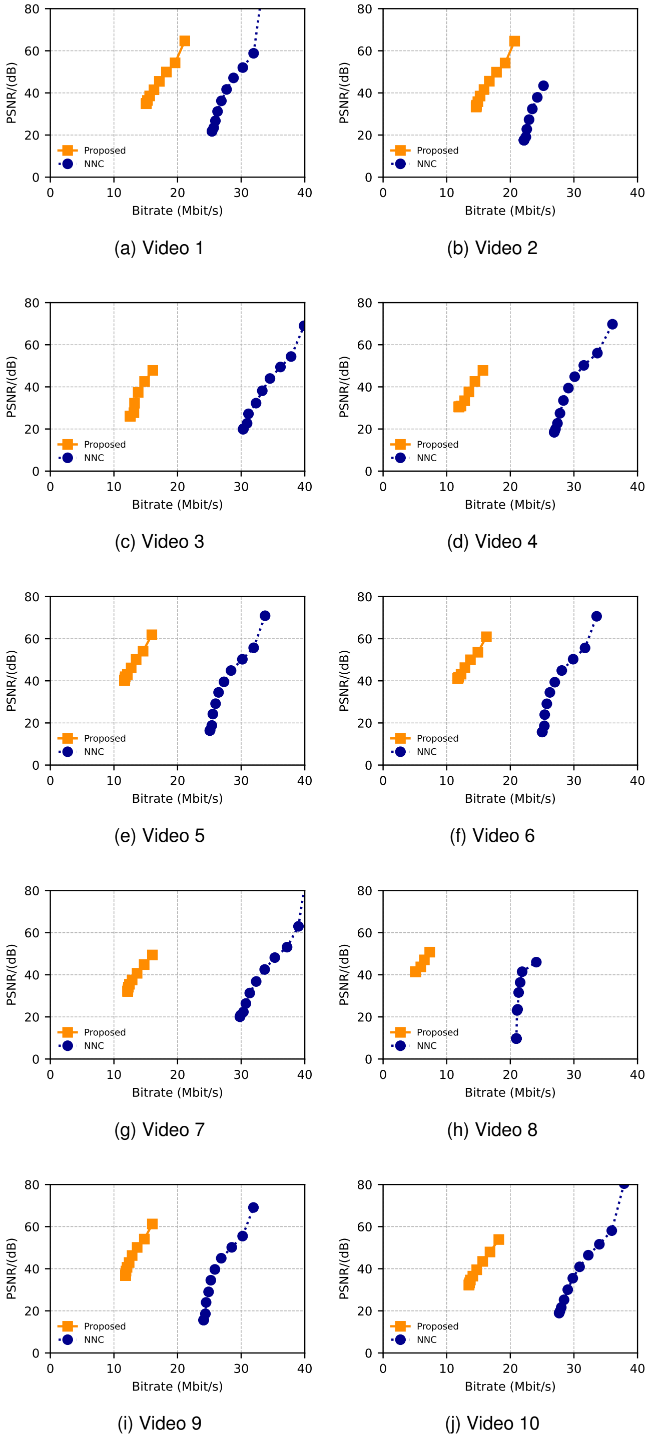
Fig. 5. Network model used for simulation: *sequential_120* represents the encoder layers and *sequential_121* represents the decoder layers of the autoencoder.

The semantic encoder is implemented using an autoencoder network with the network structure shown in 5, which creates a latent vector of dimension 1×1, which is a simplified version to evaluate performance. The system is investigated over a range of quantization levels in the residual compression network using NNCodec, and the residual neural network without using temporal prediction is compressed using NNCodec and sent to the receiver to set a benchmark for comparison of the performance.

When comparing the PSNR achieved for each quantization level, as shown in Fig. 6, it is evident that the proposed temporal prediction of the DNN parameters can achieve a better rate distortion performance compared to NNCodec in all ten videos tested. The performance for the high complexity video (Video 2) shows a marked improvement over NNCodec which does not exploit the temporal correlations between the parameters. When considering low complexity videos, such as in Videos 3 and 4, the proposed system still provides better compression, but with a smaller gain in PSNR. However, when high complexity videos are considered (such as Video 2), the PSNR gain of the proposed system at lower bit rates is significantly high, reaching 20 dB. For medium complexity videos, the PSNR gain can be observed to reach 30 dB in some cases (such as Video 8). This is due to both systems exploiting the spatial correlations between the parameters and NNCodec using a higher bitrate to achieve a better quality reconstruction of the neural network.

The proposed system adds additional complexity to the system, compared to just using NNCodec since the DNN parameter prediction has to be performed, but has significant rate distortion performance gains. The added complexity and the complexity-performance trade-off of the propsed system need to be quantified and evaluated in future research.

However, the compressed bit rates achieved by both NNCodec and the proposed system are significantly higher than those achievable by conventional video coding systems such as VVC, AOMedia Video 1 (AV1), High Efficiency Video Coding (HEVC), or Advanced Video Coding (AVC), as well as hybrids of these systems with DNN [12]. This is due to the work presented being a very basic implementation of the concept and requiring significant further refinement before it can be practically implemented to achieve performance



(a) Video 1
(b) Video 2
(c) Video 3
(d) Video 4
(e) Video 5
(f) Video 6
(g) Video 7
(h) Video 8
(i) Video 9
(j) Video 10

Fig. 6. Comparison of rate distortion performance of proposed system compared with using NNCodec alone.

on par or better than conventional video coding systems. Enhancing the temporal prediction framework and optimizing displacement vector calculation, so that displacements along a larger number of axes can be predicted, and further improving the compression rates achievable for residual compression using optimized versions of NNCodec or an alternative method are key future research areas which need to be explored.

Despite these, the proposed system is an important milestone in implementing semantic communication based media transmission systems, including video coding systems, as practical implementation of semantic communications depends on DNN and effective methods to transmit, predict or remotely train DNN parameters to minimize reconstruction losses between the transmitter and receiver is a critical capability for such systems to achieve performances on par or exceeding conventional media compression systems.

V. Conclusions

An innovative approach for solving the problem of additional overhead being required for transmitting trained decoder parameters in semantic communication based video coding systems is proposed using temporal prediction of DNN parameters to exploit the interframe correlations of video. Experimental results show rate distortion performance gains compared to using NNCodec alone, and although additional complexity is added to NNCodec by the prediction of the DNN parameters, it is compensated for by the significant gains achieved. However, performance needs further improvement to achieve rate distortion performance on par or exceeding conventional video compression standards.

References

[1] J. Adhuran, G. Kulupana, C. Galkandage, and A. Fernando, "Multiple Quantization Parameter Optimization in Versatile Video Coding for 360° Videos," *IEEE Transactions on Consumer Electronics*, vol. 66, no. 3, pp. 213–222, 8 2020.

[2] J. Adhuran, G. Kulupana, S. Blasi, and A. Fernando, "Parameter-Based Affine Intra Prediction of Screen Content in Versatile Video Coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 9, pp. 3590–3602, 9 2021.

[3] Y.-Q. Shi and H. Sun, *Image and Video Compresssion for Multimedia Engineering*. Boca Raton, FL: CRC Press, 2019.

[4] S. Han, J. Pool, J. Tran, and W. Dally, "Learning both weights and connections for efficient neural network," *Advances in neural information processing systems*, vol. 28, 2015.

[5] W. Wen, C. Wu, Y. Wang, Y. Chen, and H. Li, "Learning structured sparsity in deep neural networks," *Advances in neural information processing systems*, vol. 29, 2016.

[6] B. Bross, J. Chen, J.-R. Ohm, G. J. Sullivan, and Y.-K. Wang, "Developments in international video coding standardization after avc, with an overview of versatile video coding (vvc)," *Proceedings of the IEEE*, vol. 109, no. 9, pp. 1463–1493, 2021.

[7] D. Ding, Z. Ma, D. Chen, Q. Chen, Z. Liu, and F. Zhu, "Advances in video compression system using deep neural network: A review and case studies," *Proceedings of the IEEE*, vol. 109, no. 9, pp. 1494–1520, 2021.

[8] X. Luo, H.-H. Chen, and Q. Guo, "Semantic Communications: Overview, Open Issues, and Future Research Directions," *IEEE Wireless Communications*, vol. 29, no. 1, pp. 210–219, 2022.

[9] D. Huang, F. Gao, X. Tao, Q. Du, and J. Lu, "Toward Semantic Communications: Deep Learning-Based Image Semantic Coding," *IEEE Journal on Selected Areas in Communications*, vol. 41, no. 1, pp. 55–71, 2023.

[10] M. U. Lokumarambage, V. S. S. Gowrisetty, H. Rezaei, T. Sivalingam, N. Rajatheva, and A. Fernando, "Wireless end-to-end image transmission system using semantic communications," *IEEE Access*, vol. 11, pp. 37 149–37 163, 2023.

[11] P. Jiang, C.-K. Wen, S. Jin, and G. Y. Li, "Wireless Semantic Communications for Video Conferencing," *IEEE Journal on Selected Areas in Communications*, vol. 41, no. 1, pp. 230–244, 2023.

[12] P. Samarathunga, Y. Ganearachchi, and A. Fernando, "Video Compression by Chroma Prediction Using Semantic Communications," in *Digest of Technical Papers - IEEE International Conference on Consumer Electronics*, 2024.

[13] H. Kirchhoffer, P. Haase, W. Samek, K. Müller, H. Rezazadegan-Tavakoli, F. Cricri, E. B. Aksu, M. M. Hannuksela, W. Jiang, W. Wang *et al.*, "Overview of the neural network compression and representation (nnr) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 5, pp. 3203–3216, 2021.

[14] H. Li, A. Kadav, I. Durdanovic, H. Samet, and H. P. Graf, "Pruning filters for efficient convnets," *arXiv preprint arXiv:1608.08710*, 2016.

[15] R. Yu, A. Li, C.-F. Chen, J.-H. Lai, V. I. Morariu, X. Han, M. Gao, C.-Y. Lin, and L. S. Davis, "Nisp: Pruning networks using neuron importance score propagation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 9194–9203.

[16] N. Lee, T. Ajanthan, and P. H. Torr, "Snip: Single-shot network pruning based on connection sensitivity," *arXiv preprint arXiv:1810.02340*, 2018.

[17] P. Midtrack. People enjoying the day in a beach. www.pexels.com. Accessed: Feb. 12, 2024. [Online]. Available: https://www.pexels.com/video/people-enjoying-the-day-in-a-beach-3150419/

[18] T. Miroshnichenko. People playing soccer. www.pexels.com. Accessed: Feb. 12, 2024. [Online]. Available: https://www.pexels.com/video/people-playing-soccer-6077718/

[19] C. of Couple. A bowl of avocados and vegetables. www.pexels.com. Accessed: Feb. 12, 2024. [Online]. Available: https://www.pexels.com/video/a-bowl-of-avocados-and-vegetables-7656166/

[20] Pixabay. Blue sky video. www.pexels.com. Accessed: Feb. 12, 2024. [Online]. Available: https://www.pexels.com/video/blue-sky-video-855005/

[21] T. Elliot. A couple walking towards the launching area of the hot air balloons festival. www.pexels.com. Accessed: Feb. 12, 2024. [Online]. Available: https://www.pexels.com/video/a-couple-walking-towards-the-launching-area-of-the-hot-air-balloons-festival-3064025/

[22] D. C. Paduret. Culturing a chamomile flower plant. www.pexels.com. Accessed: Feb. 12, 2024. [Online]. Available: https://www.pexels.com/video/culturing-a-chamomile-flower-plant-3011973/

[23] S. Garenko. A girl running across a bridge. www.pexels.com. Accessed: Feb. 12, 2024. [Online]. Available: https://www.pexels.com/video/a-girl-running-across-a-bridge-19805236/

[24] V. Singh. Rangdum village in zanskar valley. www.pexels.com. Accessed: Feb. 12, 2024. [Online]. Available: https://www.pexels.com/video/rangdum-village-in-zanskar-valley-19022224/

[25] M. Kilinc. Nemrut - bitlis. www.pexels.com. Accessed: Feb. 12, 2024. [Online]. Available: https://www.pexels.com/video/nemrut-bitlis-18856748/

[26] M. T. Kirkgoz. Cold snow sea dawn. www.pexels.com. Accessed: Feb. 12, 2024. [Online]. Available: https://www.pexels.com/video/cold-snow-sea-dawn-18051870/

[27] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, Y. W. Teh and M. Titterington, Eds., vol. 9. Chia Laguna Resort, Sardinia, Italy: PMLR, 13–15 May 2010, pp. 249–256. [Online]. Available: https://proceedings.mlr.press/v9/glorot10a.html

[28] W. Fernando, C. Canagarajah, and D. Bull, "A unified approach to scene change detection in uncompressed and compressed video," *IEEE Transactions on Consumer Electronics*, vol. 46, no. 3, pp. 769–779, 2000.

[29] D. Becking, P. Haase, H. Kirchhoffer, K. Müller, W. Samek, and D. Marpe, "Nncodec: An open source software implementation of the neural network coding iso/iec standard," in *ICML 2023 Workshop Neural Compression: From Information Theory to Applications*, 2023.