Engineering Applications of Artificial Intelligence 144 (2025) 110118

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Research paper

A novel deep learning-based approach for video quality enhancement

Parham Zilouchian Moghaddam [a],[*], Mehdi Modarressi [a,b], Mohammad Amin Sadeghi [c]

[a] *School of Electrical and Computer Engineering, College of Engineering, University of Tehran, Tehran, Iran*
[b] *School of Computer Science, Institute For Research In Fundamental Sciences, Tehran, Iran*
[c] *Qatar Computing Research Institute (QCRI), Hamad Bin Khalifa University (HBKU), Doha, Qatar*

ARTICLE INFO

Keywords:
Internet of Video Things
Video compression
Auto-encoder
Deep generative models
Diffusion models

ABSTRACT

Video content has experienced a surge in popularity, asserting its dominance over internet traffic and Internet of Things (IoT) networks. Video compression has long been regarded as the primary means of efficiently managing the substantial multimedia traffic generated by video-capturing devices. Nevertheless, video compression algorithms entail significant computational demands in order to achieve substantial compression ratios. This complexity presents a formidable challenge when implementing efficient video coding standards in resource-constrained embedded systems, such as IoT edge node cameras. To tackle this challenge, this paper introduces an innovative deep-learning model specifically designed to mitigate compression artifacts stemming from lossy compression codecs. This enhancement significantly elevates the perceptible quality of low-bit-rate videos. By employing our proposed deep-learning model, the video encoder within the video-capturing node can reduce output quality, thereby generating low-bit-rate videos and effectively curtailing both computation and bandwidth requirements at the edge. On the decoder side, which is typically less encumbered by resource limitations, our suggested deep-learning model is applied after the video decoder to compensate for artifacts and approximate the quality of the original video. Experimental results affirm the efficacy of the proposed model in enhancing the perceptible quality of videos, especially those streamed at low bit rates.

1. Introduction

The boost in video content usage, driven by a demand for superior quality, has significantly amplified storage and network bandwidth needs in recent years. A study by Cisco reveals that video content comprises over 80% of global Internet traffic (Cisco, 2020). This figure unexpectedly soared during the COVID-19 pandemic, as work, education, and leisure increasingly pivoted to live video platforms. Furthermore, with numerous IoT services and applications incorporating video transmission (Bhering et al., 2022; Chen, 2020), managing video traffic has become a pivotal concern in contemporary IoT systems (Wang et al., 2017).

The immense volume of video content necessitates considerable storage space and network bandwidth, accounting for approximately 3.7% of the CO_2 emissions generated by the internet (Sarah Griffiths, 2020). This issue is particularly relevant today, as there is increasing concern about the energy consumption of computing systems and their effects on the environment and climate change. Consequently, even a modest reduction in video size can significantly decrease bandwidth usage, storage needs, and overall energy consumption. This emphasizes the essential role of video compression in enabling energy-efficient storage and transmission of video content.

Video compression, or coding, seeks to reduce the size of video files by eliminating various forms of redundancy, including statistical, temporal, and spatial redundancies within the frames. Additionally, it involves the selective omission of non-critical details to ensure that any resulting degradation in quality remains imperceptible to the human eye.

Video streaming requires a delicate balance between video quality and bit rate. Enhancing video quality typically necessitates a corresponding increase in bit rate, which, in turn, raises bandwidth requirements. Advanced compression techniques employ a range of sophisticated algorithms to effectively navigate the trade-offs between video quality and bandwidth efficiency. However, achieving a harmonious combination of low bit rate and high video quality leads to increased computational complexity for both encoders and decoders. For instance, Versatile Video Coding (VVC), one of the most recent MPEG standards, utilizes numerous coding tools and AI algorithms to achieve a 25–50% reduction in bit rate while preserving quality comparable to previous standards. Nonetheless, this innovation significantly heightens the complexities faced by encoders and decoders, increasing by factors of up to 27 times and 2 times, respectively (Chen et al., 2019).

[*] Corresponding author.
E-mail addresses: p.zilouchian@gmail.com, modarressi@ut.ac.ir, msadeghi@hbku.edu.qa (P. Zilouchian Moghaddam).

https://doi.org/10.1016/j.engappai.2025.110118
Received 8 October 2024; Received in revised form 12 December 2024; Accepted 17 January 2025
Available online 31 January 2025
0952-1976/© 2025 Elsevier Ltd. All rights reserved, including those for text and data mining, AI training, and similar technologies.

Energy consumption in encoding and decoding processes frequently escalates in tandem with computational complexity. Such elevated complexity and energy expenditure hinder the implementation of high-quality video coding standards across numerous embedded devices and IoT edge nodes, such as cameras mounted on battery-operated drones, which typically operate under stringent energy budgets and possess limited computational capabilities. Indeed, video coding in such systems must adeptly navigate the intricate balance among three competing demands: quality, bit rate, and complexity.

In this paper, we introduce NU-Class Net, a comprehensive deep learning-based technique devised to navigate the challenges of video encoding amidst constrained resources and limited network bandwidth. We have selected the name 'NU-Class Net' for our model, drawing inspiration from the iconic space shuttle featured in the "Star Wars: The Clone Wars" series. This name underscores the visual parallels between our innovative network architecture and the distinctive shape of that shuttle, highlighting our dedication to integrating creativity with advanced technology. Employing this technique, the encoder intentionally moderates video quality constraints, generating a low-bit-rate video albeit with a consequential quality degradation. Conversely, on the decoder's side, a deep learning model is utilized. This model, which is meticulously trained to counterbalance the quality loss by mitigating coding artifacts, endeavors to reconstruct the video, aspiring to attain a quality proximate to the original.

A salient feature of NU-Class Net distinguishes it from numerous contemporary works employing deep learning to devise new codecs or enhance existing ones: it does not modify or supplant a codec. Rather, it amplifies the end-to-end coding process by integrating a deep learning module subsequent to the decoder, with the aim of elevating the quality of the decoded video. Consequently, UN-Class maintains orthogonality and can be synergistically utilized with any modern video codec to mitigate the video stream bit rate further.

NU-Class Net facilitates a less resource-intensive video coding at the encoder side by involving a simplified encoding process, concurrently reducing the network bandwidth necessary for video stream transmission. On the decoder side, a deep learning model is strategically positioned subsequent to the decoder module to reconstruct video quality. The advantages of UN-Class in diminishing bandwidth and storage requisites are conspicuous. It proves exceptionally advantageous for embedded systems and IoT nodes tasked with capturing and transmitting videos under stringent energy and processing constraints. With UN-Class, these devices enjoy the luxury of less complex video encoding, obviating the need to operate intricate encoder modules that enhance quality. Correspondingly, the energy consumption for communication diminishes in proportion to the bit rate reduction. In such systems, the complexity is translocated to the decoder side, typically operating on a computer characterized by a more lenient power budget and a robust processing unit.

NU-Class Net is architecturally founded upon the U-Net, a preeminent AutoEncoder deep learning model. AutoEncoders have catalyzed breakthroughs across various computer vision tasks, especially in the realm of image reconstruction. The advent of U-Net has markedly outperformed competitors and transformed the application of encoder–decoder architectures across diverse computer vision tasks. Given that videos are composed of frames, which can be regarded as images, an opportune avenue emerges to tailor U-Net for image enhancement. NU-Class Net ingests compressed video frames as input and predicts the residual difference between the original (attainable through high-quality encoding) and compressed (low bit-rate) frames. This residual is superimposed upon the input frames to alleviate discernible encoding artifacts, thereby maintaining an output video quality that approximates the original.

Although previous research has addressed the enhancement of JPEG image quality through neural networks, NU-Class Net is purposefully designed for video applications. It takes advantage of the inter-frame correlation present in video streams to improve performance (Ledig et al., 2017; Maleki et al., 2018). Furthermore, NU-Class Net focuses exclusively on calculating the residual differences between raw and compressed frames, sidestepping the need to reconstruct an entire frame. This method effectively avoids the complexities involved in predicting a complete image with all its intricate details, resulting in improved efficiency and expedited training for the network. Our findings indicate that NU-Class Net significantly enhances the quality of low-bit-rate video frames, enabling video capture nodes to effectively utilize low-complexity, low-bit-rate video streams.

The remainder of this paper is organized into several sections that aim to enhance the reader's understanding of our research. Section 2 presents a thorough review of relevant related work, highlighting significant studies and findings in the field. In Section 3, we introduce the architecture of NU-Class Net, providing detailed insights into its innovative components and functionalities. Section 4 offers an in-depth overview of the implementation process, discussing the methodologies and technologies employed. The evaluation results are presented in Section 5, demonstrating the effectiveness and performance of NU-Class Net. Finally, Section 6 concludes the paper with a summary of our findings and their implications for future research.

2. Related work

Employing deep learning for video stream compression can be approached via various methodologies, which can be broadly classified into three primary categories as follows.

2.1. Integrating deep learning modules into codecs

One approach to augmenting various modules of existing video codecs involves the utilization of deep learning-based models. For instance, Golinski et al. (2020) introduce a novel deep-learning architecture that adeptly learns video compression in low latency mode, also discussing temporal consistency issues encountered during their experiments and proposing solutions thereto. Alternatively, Pourreza and Cohen (2021) propose a neural video codec capable of managing B-frame coding predicting a frame from both future and past reference frames as opposed to the more prevalent P-frame coding (Majumdar et al., 2004) utilized by the majority of neural encoders. This method interpolates future and past reference frames to derive a single frame, which is then employed with an existing P-frame codec, seamlessly integrating with current neural codecs that predominantly rely on the P-frame predictor. van Rozendaal et al. (2021) enhance existing neural codecs using a technique termed instance-adaptive learning, wherein they seek to modify a pre-trained compression model to transmit optimal parameters to the receiver alongside the latent code.

A predominant drawback of the aforementioned methods resides in their lack of portability; specifically, both the sender and receiver must be equipped with codecs that are deep learning-enhanced. In contrast, our approach permits the utilization of any standard codec on both ends, simply appending a neural network module subsequent to the decoder to enhance quality.

2.2. Image enhancement using deep learning

In this approach, a neural network is trained for end-to-end data compression, operating in conjunction with the codec. For instance, Maleki et al. introduced Block CNN, a novel technique designed to eliminate JPEG coding artifacts (Maleki et al., 2018). The image is partitioned into 8 × 8 pixel blocks, with the intensities of each block predicted based on preceding blocks, followed by the computation of the input image's residual. This technique repurposes JPEG's legacy processes for compression and decompression, achieving superior results compared to baselines at high compression ratios. Contrasting with Block CNN, NU-Class Net concentrates on video frames and aspires to capture lower-level features, distinguishing our work from theirs.

Ronneberger et al. pioneered the development of the U-shaped network, U-Net, designed explicitly for semantic segmentation (Ronneberger et al., 2015). The algorithm astutely abstracts compact features using a bottom-up approach, subsequently mapping them back to the original scale through a top-down branch. A notable enhancement over standard autoencoders, U-Net employs skip connections to amalgamate low-level and high-level features. The model's stellar performance has not only set a benchmark but also spawned the development of several derivatives, adept at managing more intricate image segmentation and generation tasks.

Vaccaro et al. introduced SR-UNet, a specialized U-Net model designed to enhance video quality in real-time directly on users' devices (Vaccaro et al., 2021). This neural network excels in super-resolution by reconstructing high-resolution frames from low-resolution video streams and mitigating artifacts that may occur due to video compression. SR-UNet functions as a final post-processing step, improving the visual quality of each frame before it is displayed. This allows for seamless integration with existing video coding and transmission systems without requiring modifications. However, it is essential to note that while these methods are effective, they inherently alter the image resolution and, consequently, the video file size, which may not always be desirable. In contrast, our approach is focused solely on artifact reduction without changing the resolution.

The increasing demand for multimedia processing in resource-constrained environments has spurred the development of innovative approaches to compression and quality enhancement (Noura et al., 2023; Elahi et al., 2023). Some related work try to reduce video bitrate, when the target destination of the video is a machine learning algorithm (Elahi et al., 2023). On the other hand, some related work, such as Noura et al. (2023), introduced a deep learning-based solution designed explicitly for the Multimedia Internet of Things (MIoT) context. MIoT refers to a network of interconnected devices that not only collect and transmit data but also capture and share multimedia content, such as video, audio, and images. This functionality enables devices to engage with each other through rich multimedia formats, transcending the simple sensor data found in traditional IoT systems.

Noura et al.'s approach employs Residual Dense Networks (RDN) for super-resolution and image enhancement on the application server side. This model effectively minimizes communication overhead and power consumption by achieving high compression ratios at the sender (the MIoT devices) while restoring high-quality images at the receiver's end. It specifically aims to balance visual quality with communication efficiency.

In contrast, our proposed model, NU-Class Net, takes a distinct approach to addressing similar challenges in IoT and edge-based video processing environments. Unlike the RDN-based solution proposed by Noura et al. which concentrates on image-level super-resolution and restoration, NU-Class Net operates directly on video streams. It utilizes temporal correlations to enhance the quality of individual video frames. Designed as a post-decoding enhancement module, NU-Class Net integrates seamlessly with standard codecs without altering their underlying architecture. This flexible strategy ensures easy deployment across a broad range of IoT ecosystems.

2.3. Exploring video synthesis through generative models

Recent years have witnessed the evolution of models employing residual learning, all aiming to attain superior image quality (He et al., 2016). However, a direct consequence of increasing the depth of these models (Zhang et al., 2018; Lim et al., 2017) has been a notable surge in computational complexity and memory usage.

Numerous studies investigating the application of Generative Models in video compression have primarily focused on the nuances of video encoding techniques. Among these, several significant advancements have employed Generative Adversarial Networks (GANs) to produce visually striking, photo-realistic videos from sequences of semantic maps data representations that illustrate the intended objects and their spatial relationships within a scene.

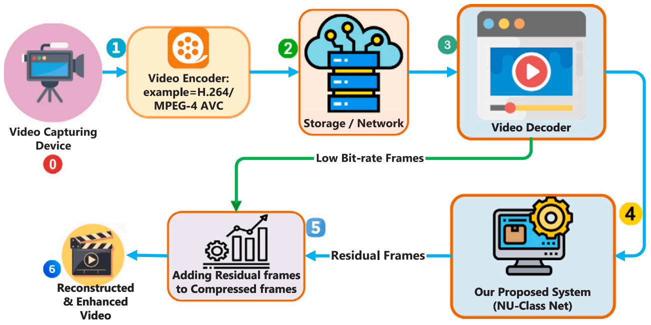
Video Capturing Device
0
1
Video Encoder: example=H.264/ MPEG-4 AVC
2
Storage / Network
3
Video Decoder
Low Bit-rate Frames
4
Our Proposed System (NU-Class Net)
Residual Frames
5
Adding Residual frames to Compressed frames
6
Reconstructed & Enhanced Video

Fig. 1. Schematic representation of the comprehensive application process of the NU-Class model.

A notable example is the Fast-Vid2Vid experiment by Zhuo et al. (2022), which demonstrates the ability of GANs to synthesize high-quality, photo-realistic video content directly from semantic maps. This work represents substantial progress in generating visually appealing video sequences based on abstract scene representations.

However, it is essential to highlight that, contrary to our approach, these GAN-based methods primarily focus on creating entirely new videos from scratch rather than enhancing the quality of existing footage. This distinction underscores the unique aspect of our research, which aims to refine and elevate previously created video material instead of simply producing new content.

2.4. Discussion

In this section, we have highlighted three key points: (1) high-level task networks, such as ResNet, excel in image classification by extracting global features; (2) generative models like GANs are specifically designed for image and video synthesis to create new content; and (3) super-resolution models, including SR-UNet and RDN, enhance low-resolution images to higher resolutions.

In contrast, NU-Class Net specializes in low-level restoration, focusing on reducing compression artifacts and enhancing degraded frames. The primary difference between NU-Class Net and other deep learning models lies in their specific applications and functionalities. NU-Class Net is carefully crafted to restore lost details and low-level features in video frames, strongly emphasizing reducing compression artifacts. This focus is essential for improving the perceptual quality of video content, particularly in situations characterized by low bit rates.

3. Proposed method

3.1. System design

Contrary to conventional approaches that primarily focus on optimizing and modifying the codec itself, we propose a nuanced solution that emphasizes enhancing the frames post-codec processing. This method diverges from traditional strategies that attempt to integrate artificial intelligence (AI) modules directly into the codec, opting instead to concentrate on improving frame quality subsequent to the codec's operation.

Fig. 1 delineates the overarching design of our system, employing the NU-Class Net model to enhance the input video quality. This technique affords a notable opportunity to compress video beyond the capacities of conventional codecs, such as HEVC (Sullivan et al., 2012). A salient advantage of our proposed method lies in its versatility; it can be appended as an extension to any video codec, irrespective of its underlying technology.

As the network depth augments and additional pooling layers are utilized to reduce the input image size, a consequential loss of lower-level features is critical, especially for our task. To counteract these

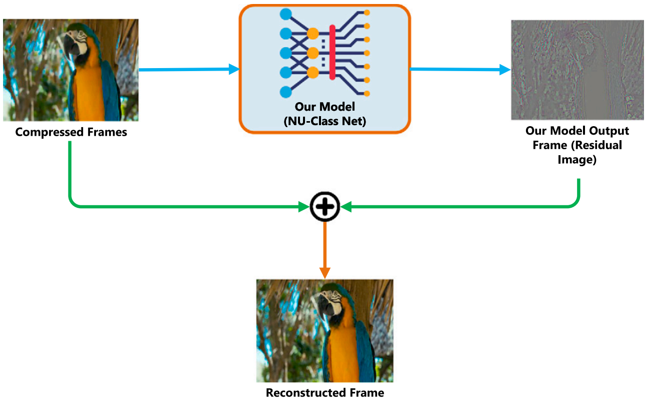

Fig. 2. How base NU-Class Net performs.

challenges, we introduce a bespoke model designed to enhance imported video quality.

The NU-Class Net processes frames received from the video encoder at reduced bit rates, producing what we refer to as the 'residual frame' for a specific input image. This residual frame represents the difference between the video frame at the reduced bit rate and its original version, all while preserving the exact resolution and aspect ratio. Once the residual frame is generated, it is added to the input image, which also has a reduced bit rate. Consequently, the final output image closely resembles the original, as illustrated in Fig. 2. A significant advantage of our approach is the substantial reduction in video file size, accomplished without modifying the underlying video codec or its related modules.

3.2. Architectural design of the NU-Class Net model

Fig. 3 outlines the architecture of NU-Class Net, embodying an encoder–decoder design paradigm. Contrary to focusing on the extraction of pivotal features for tasks like classification, our foremost objective is to derive high-level features of a video to facilitate quality enhancement. In devising the architecture of NU-Class Net, we drew inspiration from the U-Net architecture to aptly cater to this prerequisite (Ronneberger et al., 2015).

Our proposed architecture pivots around five crucial components, each integral to the enhanced functionality and performance of the NU-Class Net:

- **Encoder blocks:** The encoder section of our model is constructed of six distinct blocks, each encompassing four convolution layers. Notably, two blocks integrate pooling layers, for which we eschew traditional pooling methods in favor of convolutions with a stride of two, enhancing computational efficiency. Furthermore, we have expanded the receptive field of these blocks beyond the conventional 3 × 3 to 7 × 7 convolution windows, optimizing feature extraction (refer to Fig. 4 for the structure of each encoder block).
- **Bottleneck residual blocks (or ResBox or Residual connections):** The bottleneck segment of our network thoughtfully integrates eight residual blocks, which are instrumental in enabling the model to learn identity functions and thereby facilitate the training of a deeper neural network without incurring a consequential decrement in performance. This is actualized by adeptly mitigating issues related to both vanishing and exploding gradients (Luo et al., 2016). The formulation of a residual block is presented as (1). The structure of each implemented residual block is illustrated in Fig. 5.

$$G(x) = F(x) + x \tag{1}$$

- **Decoder blocks:** The decoder blocks symmetrically counterpoint the processes of the encoder blocks, albeit in a reversed sequence. A consistent filter size is meticulously maintained throughout the architecture. Opting for a different approach than utilized in the encoder blocks, transpose convolutions are employed in place of standard convolutions and pooling. A detailed configuration of each decoder block is illustrated in Fig. 4.
- **Skip-connections:** Skip-connections, bridging the Encoder and Decoder blocks, enable the neural network to harness both high-resolution and low-level feature information, capturing meticulous details at every pixel position—paramount for our application. This detailed information is directly conveyed to a latent layer within our network, safeguarding this invaluable data. Consequently, subsequent layers in our network are endowed with both lower resolution, high-level spatial, and contextual information, as well as low-level, texture-like information, rich in detail.
- **Final residual blocks:** Ultimately, five additional residual blocks are strategically positioned at the terminal point of our architecture, acting as guardians and conduits of indispensable information. These blocks not only preserve and transmit the most valuable data necessary for our application but also significantly mitigate the blurriness of the reconstructed frames, ensuring a sharp and clear output.

Leveraging skip-connections, originally proposed in U-Net, facilitates the extraction of higher-level and quintessential features from the data. The Encoder segment of our model is composed of six Contracting Blocks, each housing four convolution blocks. Notably, each Contracting Block is linked to the Expanding Blocks in the Decoder segment via skip-connections. The Decoder, effectively a mirrored replica of the Encoder, incorporates two inputs across its six layers: (1) The output from the preceding layer and (2) A skip connection from the corresponding block in the Encoder. Mirroring the Encoder, the Expanding Blocks within the Decoder are comprised of four convolution layers identical to those in the Contracting Block, ensuring a symmetrical and cohesive architecture.

In regards to the normalization layers, instance normalization (Ulyanov et al., 2016) was employed post each convolution layer within our network. This technique normalizes across each channel in individual training examples as opposed to normalizing across input features within a single example. Our preference, for instance, normalization, over alternatives such as batch or group normalization, is principally driven by our objective to predict the information dissipated across each channel or the imposed noise. Consequently, instance normalization emerges as a particularly advantageous strategy for our application, aligning closely with our specific concerns and operational requirements.

It merits particular note that two Feature Map Blocks have been integrated and positioned strategically at both the inception and conclusion of our entire network. These blocks shoulder the responsibility of emphasizing the convolution channels to desired channels. For example, they facilitate the modification of channel numbers, such as transmuting a four-channel input to a three-channel RGB output or vice versa, ensuring adaptability and relevance in both the input and output phases of our model.

We purposefully expanded the kernel size of the convolution windows from 3 × 3 to 7 × 7 to capture a broader range of information and improve the accuracy of shape recognition. The smaller filter size of 3 × 3 proved inadequate for our specific application, underscoring the necessity of effectively managing the receptive field. Since elements outside the receptive field do not influence the value of that unit, it is crucial to ensure comprehensive coverage of the relevant area within the frame.

It is worth noting that we have explored other kernel sizes, including 1 × 1, 5 × 5, and 9 × 9. However, each of these options presented

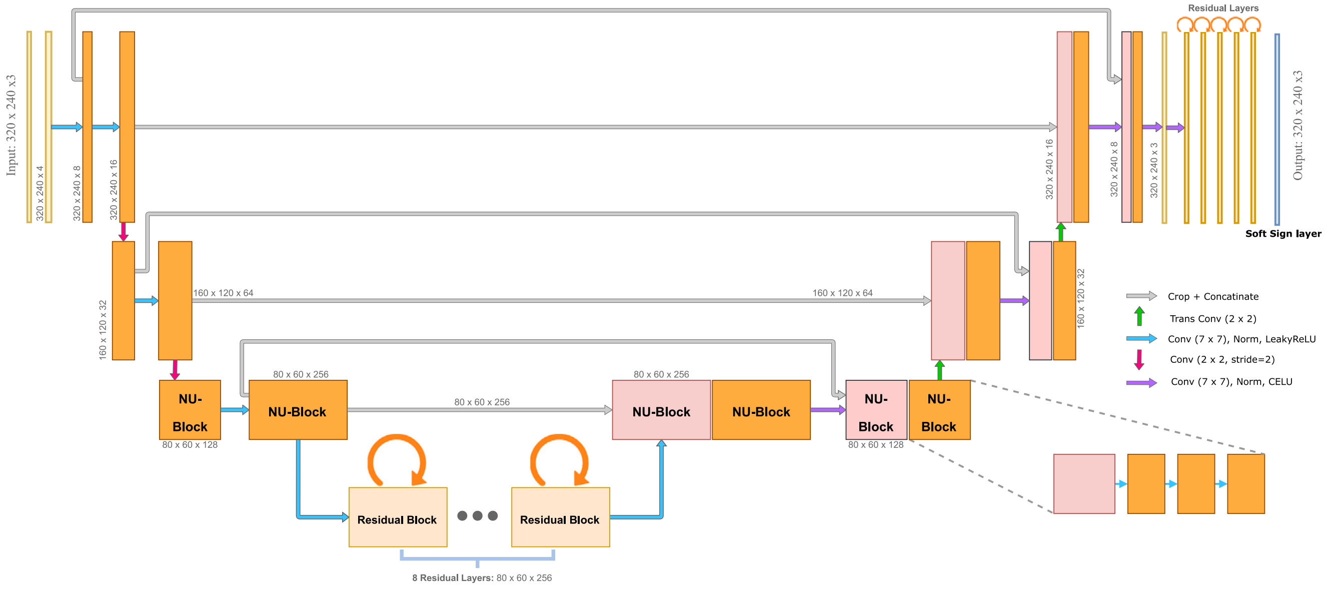

Fig. 3. Detailed architecture of the NU-Class Net, Illustrating layers, Nodes, and Connectivity patterns.

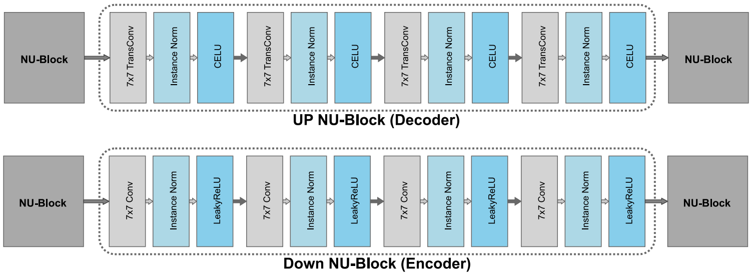

Fig. 4. Internal composition of the NU-Block, Featuring sub-components within decoder and encoder structures.

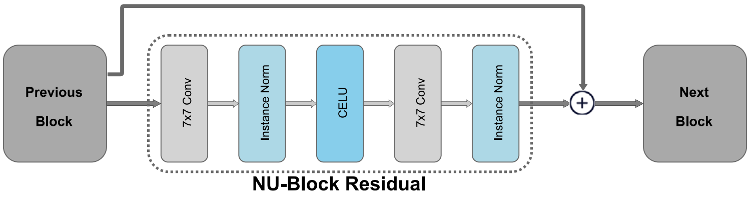

Fig. 5. Schematic illustration of the inner structure and operational flow within the NU-Block residual.

certain limitations and ultimately proved less effective than the 7 × 7 size. Our experiments have shown that the 7 × 7 kernel significantly enhances the quality of the frames compared to the alternatives. Additionally, it offers notably better execution speed, which could present a substantial advantage for our approach. This selected receptive field allows us to capture a greater variety of features, particularly lower-level ones, while maintaining efficiency during execution. In dense prediction tasks like image segmentation and optical flow estimation, specifically, our project's objective of predicting every pixel in the input image for each output pixel requires a sufficiently large receptive field to prevent the omission of critical information. Furthermore, as our project aims to enhance pixel and video frame quality, it focuses on the lower-level features present in the extracted frames. By incorporating a larger receptive field in our model, we are better positioned to capture the nuanced lower-level features within our dataset accurately.

3.3. Sequential NU-Class Net

To address the inherent challenges associated with optical flow and frame consistency – essential elements in video processing – we have introduced a novel feature in our model known as the "NU-Class Net Sequential". This feature enhances our ability to achieve improved temporal consistency while simultaneously minimizing flickering artifacts between frames. The architecture of this methodology is illustrated in Fig. 6, which outlines our strategy of utilizing residual results from preceding frames in the subsequent ones.

This innovative approach to employing prior residuals can yield subtle enhancements in video quality, particularly in action sequences and rapid frame transitions; however, it is not without its potential drawbacks. While it may marginally elevate the quality of the resultant video, the true distinction of this method lies in its training speed, requiring up to 30% fewer epochs to reach an optimal state and adequately train the network. This efficiency is rooted in the ability to leverage knowledge obtained from earlier frames to predict those that follow.

During the training phase, the model's output for frame t is combined with the compressed frame $t + 1$. This merged frame is then fed into the model to predict its residual frame, serving as the foundation for the model's training. It is worth noting that, although this process

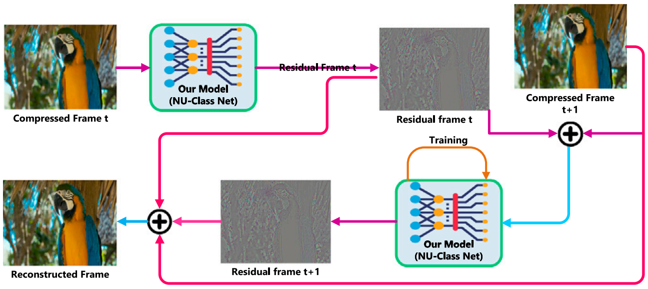

Fig. 6. Architecture and training workflow of the sequential NU-Class Net model.

remains consistent during the inference phase, the model ceases to train at that time.

3.4. *Diffusion NU-Class Net*

We present the Diffusion NU-Class Net, which utilizes the fundamental principles of diffusion models (Song et al., 2020). This innovative approach is designed to systematically remove added noise from images through a series of iterative and progressive processes. To predict the frame's residual, we employ 'three' NU-Class Net models, implementing a three-stage prediction process. The selection of three stages represents a strategic compromise between optimizing runtime and enhancing the quality of the results. A comprehensive illustration of the network architecture and its training methodology can be found in Fig. 7.

During the training phase, a progressive and consecutive-like approach is employed, beginning with the first model. Upon completion of its training, the first model serves as a foundation for training the second model. Following this, both the first and second trained models collaborate to support the training of the third model. For inference, all three trained models are utilized in succession, steadily enhancing the quality of the input frame at each stage.

3.5. *Exploring generalizability in NU-Class Net*

The distinguishing feature of our approach is its impressive generalization capability, which sets it apart from other methods. NU-Class Net is able to perform effectively not only on the videos it was trained on but also on those it has never encountered before. This unique characteristic is a result of the model's focus on learning intricate, fine-grained lower-level features of video frames, rather than merely concentrating on high-level contextual information. Consequently, the model achieves satisfactory performance even with unfamiliar videos, leveraging its lower-level features and its foundational design aimed at enhancing the quality of unseen videos. It is essential to recognize that the generalization performance of the model is significantly influenced by the training dataset and the diversity of the content it encompasses.

4. Implementation

In the implementation of our approach, images of dimensions 240 × 320 were utilized as video frames. The network was trained using NVIDIA GPUs, specifically Tesla P100 (16 GB) (Nvidia, 2016), V100 (16 GB) (Nvidia Tesla, 2020), and A100 (40 GB) (Nvidia A100, 2022), within Google Colab Pro and Pro+ environments. The training was conducted over a span of 200 epochs to ensure robust learning and convergence. Our model, comprising 79,975,939 parameters, is structured with 12 NU-Blocks and 13 NU-Block Residuals, meticulously designed to balance complexity and computational efficiency.

To expedite and enhance the training process, we leveraged schedulers for learning rate adjustment, enabling a dynamic reduction in the learning rate contingent upon specified validation measurements, and applied post-optimizer update. The "ReduceLROnPlateau" scheduler was utilized, which judiciously diminishes the learning rate upon detecting a plateau in the chosen performance metric, herein, the validation loss. This strategy is intended to judiciously modify the learning rate in alignment with the observed stagnation in performance metrics, facilitating a more nuanced adaptation throughout the learning phase.

In the model's learning process, Adam (Kingma and Ba, 2014), was employed as the optimizer, renowned for its efficacy in various deep learning applications. Furthermore, the loss function utilized for our model was Pixel-Distance Loss (Isola et al., 2017), which leverages the Mean Absolute Error (MAE, L1Loss) metric. The formulations for MAE, Mean Squared Error (MSE), and Pixel Loss are provided as Eqs. (2), (3), and (4), respectively.

$$\ell(x, y) = \{l_1, \ldots, l_N\}^T,\ l_n = |\, x_n - y_n \,|\ (MAE\ Loss) \tag{2}$$

$$\ell(x, y) = \{l_1, \ldots, l_N\}^T,\ l_n = (x_n - y_n)^2\ (MSE\ Loss) \tag{3}$$

$$\ell(x, y) = \{l_1, \ldots, l_N\}^T = \lambda * \Sigma |\, x_n - y_n \,|\ (Pixel\ Loss) \tag{4}$$

We opted for this particular loss function over MSE Loss due to the pixel-wise differences under examination; each of these differences is less than one. The Mean Absolute Error (MAE) proves superior in capturing these discrepancies compared to the MSE as it computes the average absolute difference, providing a measure that is more robust to outliers. Consequently, MAE facilitates a more adept handling of minor variations, thereby optimizing the utility of our model by adeptly managing nuanced differences.

A key aspect of our approach to loss calculation is the adoption of "Residual Loss" in place of traditional loss metrics. This method aims to assess the difference between the output of our model and the Pixel Distance Loss that arises between raw and compressed frames. By leveraging this innovative strategy, our model predicts the residuals of the compressed frames rather than the frames themselves. This predicted residual is then added to the compressed frame, enhancing its overall quality. This technique adeptly addresses the challenges encountered in processing video frames, such as rapid scene transitions and diverse video compositions, by circumventing potential issues that

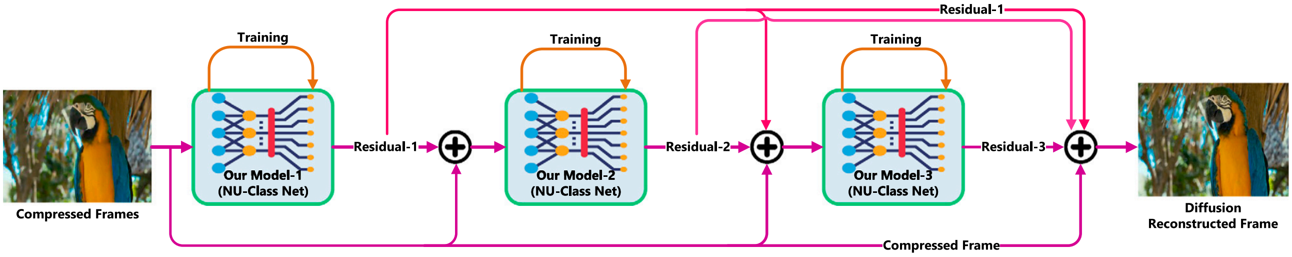

Fig. 7. Visualization of the training process and practical implementation of the diffusion NU-Class Net, Rooted in diffusion model principles.

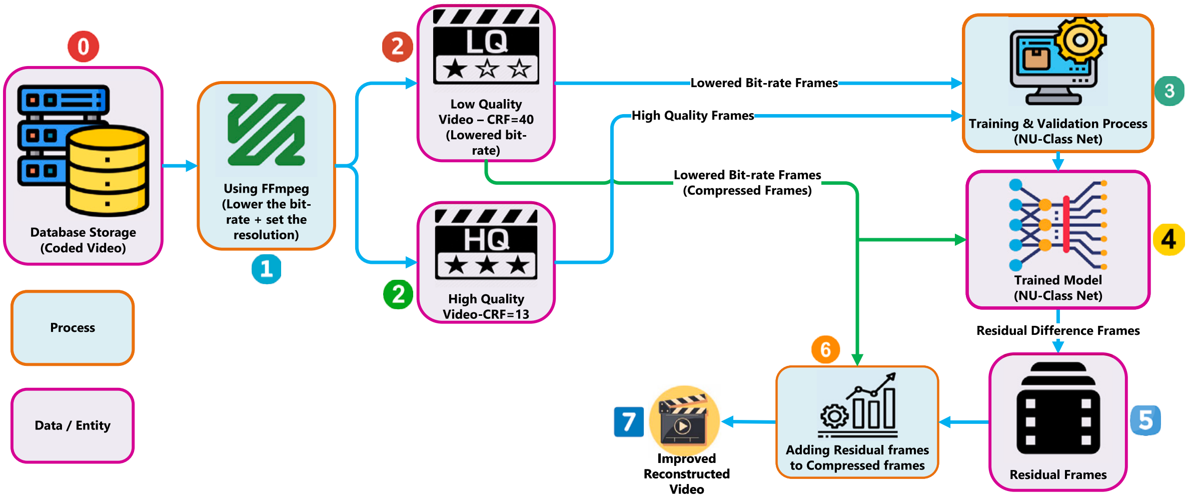
0
Database Storage (Coded Video)
1
Using FFmpeg (Lower the bit-rate + set the resolution)
2
LQ
Low Quality Video – CRF=40 (Lowered bit-rate)
2
HQ
High Quality Video-CRF=13
Lowered Bit-rate Frames
High Quality Frames
Lowered Bit-rate Frames (Compressed Frames)
3
Training & Validation Process (NU-Class Net)
4
Trained Model (NU-Class Net)
Residual Difference Frames
5
Residual Frames
6
Adding Residual frames to Compressed frames
7
Improved Reconstructed Video
Process
Data / Entity

Fig. 8. Schematic overview of the seven-stage training process applied to all three variants of the NU-Class Net.

Table 1
Comparative properties of raw and compressed videos, including metrics such as bitrate, frame rate, and resolution.

Video type	CRF	Bit rate (kb/s)	Resolution	Color space	Bit depth	File size (MB)
Raw Video	13	3572	320 × 240	RGB	8-bit	7910
Compressed Video	40	235	320 × 240	RGB	8-bit	445.7
Typical Video	18	1947	320 × 240	RGB	8-bit	4652.6

may result from these factors (see Fig. 8).

A pivotal aspect warranting discussion pertains to the dataset utilized in our study. Given the absence of a pre-existing dataset specifically tailored to this distinctive task and devoid of analogous applications, we curated our own. This was accomplished by utilizing a video file approximately 2 h and 15 min in length, with a frame rate of 30 FPS. Subsequently, frame extraction was performed using the ubiquitously employed FFmpeg tool (Anon, 2025; Tomar, 2006), which affords the capability of extracting video frames at any designated frame rate. For the construction of our dataset, six frames per second were selected. These frames were extracted under two distinct conditions: low quality and high quality. Within FFmpeg, the Constant Rate Factor (CRF) value determines the quality of the output video, with a lower CRF yielding higher quality, albeit at an elevated bit rate. A CRF value ranging from 17 to 23 is conventionally deemed a judicious balance between quality and rate. In our work, we elected to use CRF values of 13 and 40 for raw and low-bit-rate videos, respectively.

Approximately 15 min of our video were allocated for testing purposes, with the remainder being dedicated to training.

5. Experiments & results

In this section, we critically evaluate the performance of the proposed network, employing three distinct metrics: Pixel-Distance Loss, Peak Signal-to-Noise Ratio (PSNR), and Structural Similarity Index Measure (SSIM). Subsequent discussions will compare and interpret the derived results, providing a comprehensive analysis of the network's efficacy and robustness.

Initially, we delve into the results derived from the Pixel-Distance Loss metric. As illustrated in Fig. 9, a conspicuous enhancement in the quality of the reduced bit-rate frames is achieved using the NU-Class Net.

Centering our attention on the Pixel Distance Loss, this metric elucidates the disparities between pixels within every corresponding frame in the dataset. Moreover, as delineated in Fig. 9, there is a discernible reduction in the loss of the test set, approximately 30 percent, upon the conclusion of the training process.

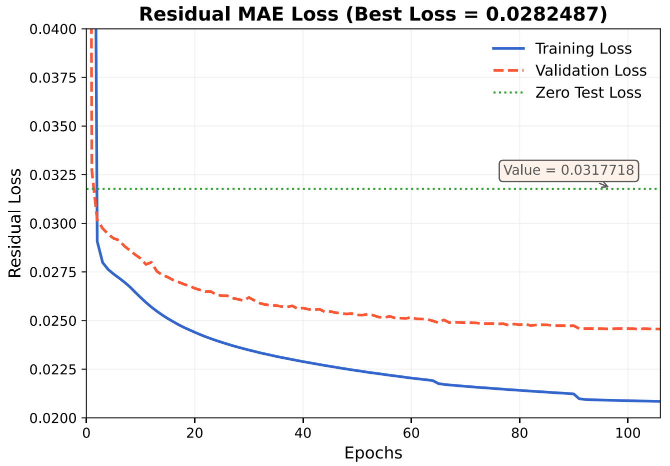

Fig. 9. The loss curve of the basic NU-Class Net model, showing the training and testing loss for each epoch. This visualization provides insight into the model's performance and convergence over time.

Table 1 presents the characteristics of both the compressed and raw videos employed in our model, all at a resolution of 320 × 240. This resolution was selected for two key reasons: first, the resource requirements for training our model with larger frames exceeded our available capacity. Second, the potential storage demands associated with larger resolutions would have posed a considerable challenge. It is worth noting that, given the exploratory nature of our project, the resolution is not critical for validating our proposed approach.

As shown in Table 1, our compressed video files have achieved a remarkable size reduction of up to 17 times while preserving the original resolution. This means that, with identical resolution and aspect ratio, our compressed videos occupy only $\frac{1}{17}$ of the storage space required by the original files. This significant reduction in size allows for additional content to be stored within the same storage capacity without compromising the initial quality. Given that our model is

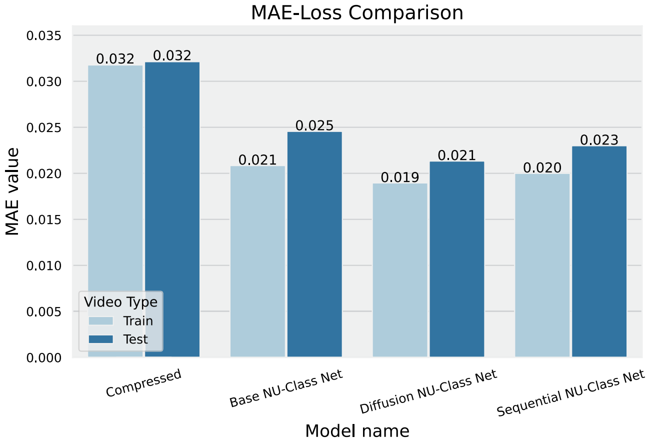

Fig. 10. The final MAE loss comparison.

designed to enhance the quality of the compressed videos, we anticipate an improvement in video quality as a result.

Furthermore, Table 1 details the characteristics of what we classify as a ‘typical video’. This category of video experiences a certain degree of lossiness that does not significantly detract from the user experience. A comparison of the attributes of a compressed video against this typical benchmark reveals that our method achieves an approximate tenfold reduction in size. This remarkable compression rate is preserved even under conditions where video quality is essential, thus underscoring the effectiveness of our approach.

5.1. Evaluating loss

The terminal Mean Absolute Error (MAE) loss achieved is depicted in Fig. 10. Notably, the proposed approach has precipitated a significant diminution in the loss term. This shows that we have been able to diminish lots of the added noise to the frames due to the drop in the bitrate of the video.

5.2. Evaluating PSNR and SSIM

Regarding the Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM), our video was reconstructed using the revitalized frames, followed by the computation of both evaluation metrics. Fig. 11 demonstrates our achievement in meeting both SSIM and PSNR criteria by elevating the results from an unacceptable quality measure to an acceptable threshold. Specifically, acceptable PSNR values must exceed 30, a benchmark successfully met by all three proposed models. Furthermore, we have elevated SSIM results from below the acceptable threshold of 0.9 to above it, thereby aligning them within the appropriate quality range for SSIM.

Our video was reconstructed using the revitalized frames, followed by the computation of both evaluation metrics. PSNR and SSIM are widely used metrics to quantitatively assess the quality of reconstructed images and videos. Specifically, SSIM evaluates the similarity between two images by assessing changes in structural information, luminance, and contrast. It provides a perceptual metric that correlates well with human visual perception. Higher SSIM values indicate a greater similarity to the original image, with acceptable values typically exceeding 0.9. The SSIM between two image patches, x and y, is calculated as:

$$\mathrm{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}, \tag{5}$$

where:

- μ_x and μ_y: mean pixel intensities of x and y,
- σ_x^2 and σ_y^2: variances of x and y,
- σ_{xy}: covariance between x and y,
- $C_1 = (K_1 L)^2$ and $C_2 = (K_2 L)^2$: constants to stabilize the division, where L is the dynamic range of pixel values (e.g., 255 for 8-bit images), and K_1 and K_2 are typically set to 0.01 and 0.03, respectively.

SSIM values range from -1 to 1, with values closer to 1 indicating higher similarity.

In contrast, PSNR quantifies the peak error between two images and serves as a simple yet effective measure of image quality. It is defined using the logarithm of the ratio between the maximum possible power of a signal and the power of the corrupting noise. Higher PSNR values generally indicate better quality, with values above 30 considered acceptable. The PSNR is calculated as:

$$\mathrm{PSNR} = 10 \cdot \log_{10}\left(\frac{\mathrm{MAX}_I^2}{\mathrm{MSE}}\right), \tag{6}$$

where:

- MAX_I: maximum possible pixel value of the image (e.g., 255 for an 8-bit image),
- MSE: Mean Squared Error between the original image I and the reconstructed image K, calculated as:

  $$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}(I_i - K_i)^2, \tag{7}$$

  where I_i and K_i are the pixel values at the ith location in the original and reconstructed images, respectively, and N is the total number of pixels.

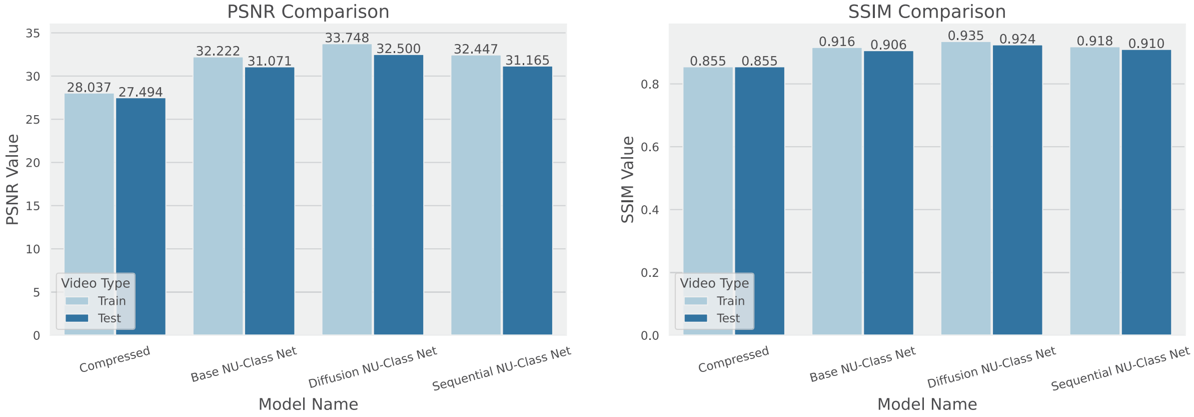

Fig. 11. Comparative analysis of PSNR and SSIM values across all proposed models, Distinguished between train and test frames.

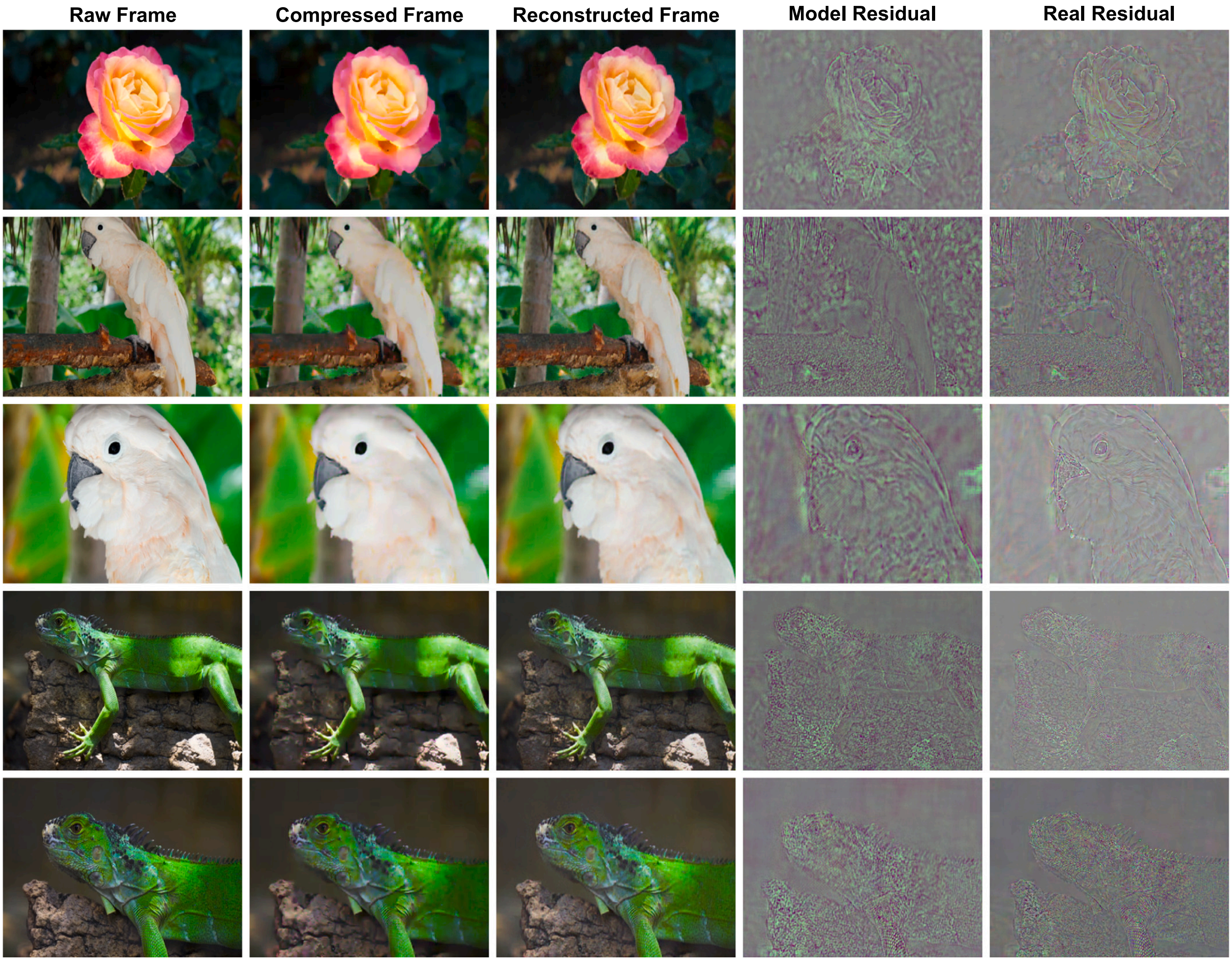

Fig. 12. Visualization of frames reconstructed by the NU-Class Net, Demonstrating fidelity and accuracy in reproduction.

Fig. 11 highlights our success in fulfilling both the SSIM and PSNR criteria by enhancing the results from an unacceptable quality level to an acceptable standard. To be considered acceptable, PSNR values must exceed 30, a benchmark that all three proposed models have successfully achieved. Furthermore, we have improved the SSIM results from below the acceptable threshold of 0.9 to above it, thus placing them firmly within the appropriate quality range for SSIM.

Fig. 12 exhibits a qualitative comparison across raw, compressed, and reconstructed frames, illustrated through five distinct frames within the test dataset. The figure presented here offers a distinct visualization of the effectiveness of our model in practical scenarios, particularly in enhancing the quality of video frames. It demonstrates the model's proficiency in predicting the residual of the frames, which represents the discrepancy between the original and reduced bitrate frames. This capability is shown to achieve a commendable level of accuracy in real-world applications. As depicted in Fig. 12, the model notably excels in restoring and augmenting the lost quality of video frames, achieving a high standard of output quality.

5.3. Comparison of three different proposed methods for NU-Class Net

We have introduced three distinct versions of NU-Class Net: Base NU-Class Net, NU-Class Net Sequential, and Diffusion NU-Class Net. Each version presents its own set of advantages and disadvantages. As illustrated in Figs. 10 and 11, the Diffusion NU-Class Net demonstrates the most notable improvements in both frame quality and loss. Following closely, the Sequential NU-Class Net ranks second in terms of quality enhancement, while the Base NU-Class Net falls in last place.

However, both the Diffusion NU-Class Net and the Sequential NU-Class Net encounter limitations in processing speed and parallel execution. In this regard, the Base NU-Class Net excels in performance. The Diffusion NU-Class Net operates at roughly one-third the speed of the Base NU-Class Net due to its requirement for the sequential execution of three different networks, which significantly impairs its operational speed.

On the other hand, the Sequential NU-Class Net improves processing efficiency by leveraging predicted residual frames from previous video frames, particularly during the training phase. While this approach aids in achieving faster convergence in the training process when compared to the Base NU-Class Net, it still suffers a decrease in speed. This is because the Sequential NU-Class Net must wait for the residual calculations of each frame before proceeding, further constraining its ability for parallel execution.

5.4. Execution time analysis

Execution time is a crucial factor in deploying NU-Class Net. As the Constant Rate Factor (CRF) increases at the edge node, both the quality and bit rate of the video decrease. This reduction in quality lowers the execution time, which subsequently decreases the computational power requirements at the edge nodes.

Our assessments indicate that changing the CRF from a baseline value of 18 to 40 results in a 63 percent reduction in the encoder's execution time. The primary reason for this decreased complexity is that lowering the video quality increases the step size of quantization, which in turn reduces the precision of the transformed data. This allows the encoder to simplify its prediction and coding processes, such as motion estimation and transformation, thereby reducing processing requirements.

On the decoder side, the reduced bit rates mean there are fewer bits to process, which leads to a faster decoding time and a lower processing load on the system. However, we introduce the complexity of a proposed deep learning model at the decoder stage.

Although the NU-Class Net model primarily operates on devices with fewer resource constraints compared to edge-based nodes, it is still essential to complete video frame processing within a reasonable time frame. Our empirical evaluation reveals that the execution time of the NU-Class Net, using previously defined parameters, on a V100 GPU for a single frame is 83.33 ms, enabling video processing at a rate of 24 frames per second.

To further reduce execution time, we can utilize dedicated hardware accelerators (Daneshtalab and Modarressi, 2020) or implement software-based optimization and acceleration methods, as suggested in our prior work (Seyedolhosseini et al., 2021; Khodarahmi et al., 2024). Currently, we employ basic quantization by converting parameters and feature maps from 32-bit floating-point to 16-bit fixed-point numbers.

Observations indicate that reducing the precision to 14-bit fixed-point numbers maintains output accuracy within one percent of the original. Consequently, the model operates with 16-bit fixed-point numbers, which is the closest word size to 14-bit supported by our platform, effectively reducing the model size by approximately 2× and lowering the execution time to 19.84 ms. It is important to note that quantization is one of the more straightforward acceleration techniques; more advanced acceleration methods will be explored in future work.

6. Conclusion

In the current work, we have introduced NU-Class Net, a groundbreaking deep learning model engineered to mitigate coding noise within low-bit-rate compressed videos. Capitalizing on the robust architecture of NU-Class Net, our approach adeptly revitalizes a myriad of details that are frequently lost during video compression, without necessitating the creation of new or alterations to existing video codecs. Instead, our model strategically operates post-decoding, utilizing a conventional codec to recover the intricate video details that tend to be compromised by a low-bit-rate encoder. The capabilities offered by our method are extensive, facilitating seamless integration with a wide array of existing video codecs. Evidenced by up to a 40% improvement in perceivable video quality, as indicated by MAE Loss on our utilized benchmarks, NU-Class Net stands out as a significant contribution to the field.

The outcomes of our research serve not merely as a validation but as a tangible proof of concept, illustrating the feasibility of employing deep learning models to facilitate a high-quality video experience in IoT systems, particularly those constrained by limited computational power and bandwidth at the edge nodes. Looking forward, future research endeavors will necessitate the amplification of the neural network model's performance potentially through the infusion of additional video-specific capabilities and refined training methodologies and a reduction in its complexity, which may be achieved via the application of hardware acceleration. This progression will serve to further cement the applicability and efficiency of deep learning models in optimizing video quality within constrained IoT environments.

CRediT authorship contribution statement

Parham Zilouchian Moghaddam: Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Mehdi Modarressi:** Writing – review & editing, Validation, Supervision, Resources, Project administration, Conceptualization. **Mohammad Amin Sadeghi:** Validation, Supervision, Resources, Project administration.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Parham Zilouchian Moghaddam reports was provided by University of Tehran. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

Anon, 2025. [Online]. Available: https://ffmpeg.org/.

Bhering, F., Passos, D., Ochi, L.S., Obraczka, K., Albuquerque, C., 2022. Wireless multipath video transmission: when IoT video applications meet networking—a survey. Multimedia Syst. 28 (3), 831–850.

Chen, C.W., 2020. Internet of video things: Next-generation IoT with visual sensors. IEEE Internet Things J. 7 (8), 6676–6685.

Chen, J., Ye, Y., Kim, S., 2019. Algorithm description for versatile video coding and test model 1 (VTM 1). In: Joint Video Experts Team (JVET) of ITU-T SG. Vol. 16, pp. 10–20.

Cisco, U., 2020. Cisco Annual Internet Report (2018–2023) White Paper. Cisco, San Jose, CA, USA.

Daneshtalab, M., Modarressi, M., 2020. Hardware Architectures for Deep Learning. Institution of Engineering and Technology (IET) Pub.s.

Elahi, A., Falahati, A., Pakdaman, F., Modarressi, M., Gabbouj, M., 2023. NCOD: Near-optimum video compression for object detection. In: 2023 IEEE International Symposium on Circuits and Systems. ISCAS, pp. 1–5.

Golinski, A., Pourreza, R., Yang, Y., Sautiere, G., Cohen, T.S., 2020. Feedback recurrent autoencoder for video compression. In: Proceedings of the Asian Conference on Computer Vision.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 770–778.

Isola, P., Zhu, J.-Y., Zhou, T., Efros, A.A., 2017. Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1125–1134.

Khodarahmi, M., Modarressi, M., Elahi, A., Pakdaman, F., 2024. ReMove:Leveraging motion estimation for computation reuse in CNN-based video processing. In: 2024 5th CPSSI International Symposium on Cyber-Physical Systems (Applications and Theory). CPSAT, pp. 1–7.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al., 2017. Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4681–4690.

Lim, B., Son, S., Kim, H., Nah, S., Mu Lee, K., 2017. Enhanced deep residual networks for single image super-resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 136–144.

Luo, W., Li, Y., Urtasun, R., Zemel, R., 2016. Understanding the effective receptive field in deep convolutional neural networks. Adv. Neural Inf. Process. Syst. 29.

Majumdar, A., Ramchandran, K., Tagliasacchi, M., et al., 2004. A distributed-source-coding based robust spatio-temporal scalable video codec. In: Picture Coding Symposium. PCS 2004, pp. 473–478.

Maleki, D., Nadalian, S., Mahdi Derakhshani, M., Amin Sadeghi, M., 2018. Blockcnn: A deep network for artifact removal and image compression. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 2555–2558.

Noura, H., Azar, J., Salman, O., Couturier, R., Mazouzi, K., 2023. A deep learning scheme for efficient multimedia IoT data compression. Ad Hoc Netw. 138 (15), 102998, [Online]. Available: https://hal.science/hal-04224626.

2016. Nvidia tesla p100: The most advanced data center accelerator. [Online]. Available: https://www.nvidia.com/en-us/data-center/tesla-p100/.

2022. Nvidia A100 gpus power the modern data center. [Online]. Available: https://www.nvidia.com/en-us/datacenter/a100/.

2020. Nvidia tesla v100. [Online]. Available: https://www.nvidia.com/en-gb/data-center/tesla-v100/.

Pourreza, R., Cohen, T., 2021. Extending neural p-frame codecs for b-frame coding. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6680–6689.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 234–241.

van Rozendaal, T., Brehmer, J., Zhang, Y., Pourreza, R., Cohen, T.S., 2021. Instance-adaptive video compression: Improving neural codecs by training on the test set. arXiv preprint arXiv:2111.10302.

Sarah Griffiths, 2020. Why your internet habits are not as clean as you think. [Online]. Available: https://www.bbc.com/future/article/20200305-why-your-internet-habits-are-not-as-clean-as-you-think.

Seyedolhosseini, A., Modarressi, M., Masoumi, N., Karimian, N., 2021. Efficient photodetector placement for daylight-responsive smart indoor lighting control systems. J. Build. Eng. 42, 103013.

Song, Y., Sohl-Dickstein, J., Kingma, D.P., Kumar, A., Ermon, S., Poole, B., 2020. Score-based generative modeling through stochastic differential equations. arXiv preprint arXiv:2011.13456.

Sullivan, G.J., Ohm, J.-R., Han, W.-J., Wiegand, T., 2012. Overview of the high efficiency video coding (HEVC) standard. IEEE Trans. Circuits Syst. Video Technol. 22 (12), 1649–1668.

Tomar, S., 2006. Converting video formats with FFmpeg. Linux J. 2006 (146), 10.

Ulyanov, D., Vedaldi, A., Lempitsky, V., 2016. Instance normalization: The missing ingredient for fast stylization. arXiv preprint arXiv:1607.08022.

Vaccaro, F., Bertini, M., Uricchio, T., Del Bimbo, A., 2021. Fast video visual quality and resolution improvement using SR-UNet. In: Proceedings of the 29th ACM International Conference on Multimedia. pp. 1221–1229.

Wang, W., Wang, Q., Sohraby, K., 2017. Multimedia sensing as a service (MSaaS): Exploring resource saving potentials of at cloud-edge IoT and fogs. IEEE Internet Things J. 4 (2), 487–495.

Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., Fu, Y., 2018. Image super-resolution using very deep residual channel attention networks. In: Proceedings of the European Conference on Computer Vision. ECCV, pp. 286–301.

Zhuo, L., Wang, G., Li, S., Wu, W., Liu, Z., 2022. Fast-Vid2Vid: Spatial-temporal compression for video-to-video synthesis. In: European Conference on Computer Vision. Springer, pp. 289–305.