

Video Quality Enhancement with Deep Neural Networks on Edge Devices: Architectures, Optimizations, and Challenges

Garibaldi da Silveira Júnior
garibaldi.dsj@inf.ufpel.edu.br
ViTech - PPGC - UFPel
Pelotas, Brazil

Gilberto Kreisler
gkfneto@inf.ufpel.edu.br
ViTech - PPGC - UFPel
Pelotas, Brazil

Bruno Zatt
zatt@inf.ufpel.edu.br
ViTech - PPGC - UFPel
Pelotas, Brazil

Daniel Palomino
dpalomino@inf.ufpel.edu.br
ViTech - PPGC - UFPel
Pelotas, Brazil

Guilherme Correa
gcorrea@inf.ufpel.edu.br
ViTech - PPGC - UFPel
Pelotas, Brazil

ABSTRACT

The growing demand for high-quality video content and the dominance of video traffic in modern networks have intensified the need for efficient video quality enhancement methods capable of operating on edge devices. Although deep neural networks have achieved state-of-the-art performance in tasks such as compression artifact reduction, denoising, and video super-resolution, most existing architectures rely on computationally expensive operations that hinder real-time deployment on resource-constrained platforms. This paper presents a systematic review of deep neural network based video quality enhancement architectures designed for edge devices, with a focus on real-time performance and efficiency–quality trade-offs. Existing approaches are analyzed by categorizing them into system-level video delivery frameworks and optimized neural models, highlighting key architectural choices, optimization strategies, and validation practices. The review reveals common trends, limitations, and open challenges, emphasizing that effective VQE on edge devices requires carefully balancing perceptual quality, computational complexity, memory usage, and energy consumption.

KEYWORDS

Video Quality Enhancement, Deep Neural Networks, Edge Devices, Real-Time Processing

1 INTRODUCTION

Video consumption has accounted for the largest share of global Internet data traffic in recent years, driven primarily by the widespread adoption of video-on-demand services, live streaming, and social media platforms. Recent reports indicate that video traffic represents more than 70% of the total volume of data transmitted worldwide, a trend that continues to grow steadily [9, 35].

In this context, video compression plays a central role for companies whose activities are directly related to the storage, distribution, and delivery of audiovisual content. Efficient video coding technologies enable the transmission of high-quality video at reduced bitrates, supporting service scalability, lowering operational costs,

and improving the user experience under diverse network conditions.

Furthermore, the increasing demand for higher-resolution content, such as Ultra High Definition (UHD) and 4K video, significantly contributes to the growth in the amount of data required for video transmission and storage. Despite continuous advances in compression standards, including High Efficient Video Coding (HEVC), AV1, and the more recent Versatile Video Coding (VVC) [1], the rise in spatial resolution and frame rates imposes additional challenges on network infrastructure and end-user devices [29].

Given this scenario, it becomes essential to investigate strategies capable of reducing the volume of transmitted data without compromising perceptual video quality, seeking a balance between coding efficiency and the quality of experience perceived by end users.

The field of Video Quality Enhancement (VQE) aims to improve the Quality of Experience (QoE) perceived by end users by acting as a complementary or post-processing stage to video compression. VQE techniques include compression artifact reduction [49], which seeks to mitigate distortions such as blocking, ringing, and blurring introduced by video codecs; denoising [10], which focuses on removing noise present before or after compression; and Video Super-Resolution [15] (VSR), whose objective is to artificially reconstruct higher-resolution versions from low-resolution signals while preserving visual fidelity as much as possible.

Figure 1 illustrates the complete pipeline of video compression and quality enhancement. In step (a), the original high-quality video is composed by frames captured by a camera device in pixel format, typically in the RGB color space, where each pixel is represented by three color channels: red, green, and blue. In step (b), the video undergoes the compression process, where the encoder converts the original signal into a compressed bitstream to reduce bandwidth and storage requirements. After transmission or storage, the decoder reconstructs the compressed content, producing the decoded video shown in step (c). At this stage, the decoded video may contain visible compression artifacts, including blocking, ringing, blurring, and loss of fine spatial details. The severity of these distortions depends on factors such as the video codec, compression parameters, content characteristics, and target bitrate.

To mitigate these degradations, step (d) applies a VQE model directly on the edge device as a post-processing stage. The objective of the VQE model is to improve the perceptual quality of

the decoded video while respecting the computational, memory, and energy constraints of real-time edge platforms. Finally, step (e) represents the enhanced video generated after the reconstruction process, with reduced artifacts and improved visual quality compared to the decoded input. Depending on the target task, the enhanced video may also present a higher spatial resolution than the decoded input, as commonly observed in video super-resolution approaches.

For VQE, models based on Deep Neural Networks (DNNs), particularly those formulated as image and video reconstruction problems, have demonstrated state-of-the-art performance in both objective and perceptual quality metrics. Modern architectures for video super-resolution and restoration have shown the ability to reconstruct content at resolutions such as 4K from lower-resolution inputs, achieving quality comparable to videos encoded directly at higher resolutions, but at significantly reduced bitrates [23].

However, despite the high performance achieved by these models, a large portion of the architectures proposed in the literature exhibits high computational complexity, substantial memory consumption, and a strong dependence on high-performance accelerators. These characteristics hinder their real-time deployment on resource-constrained portable devices, such as smartphones and integrated GPUs [18, 43]. Real-time performance is typically achieved when the model can process more than 24 Frames Per Second (FPS). This limitation has motivated research efforts and competitive benchmarks focused on developing video super-resolution models capable of operating on mobile platforms with acceptable latency and energy consumption.

Consequently, the primary challenge in designing VQE architectures for edge devices lies in identifying an appropriate trade-off between objective and perceptual quality, computational cost, energy consumption, and memory constraints, in order to enable real-time execution under strict resource limitations [16, 28, 30, 33, 36, 55]. In this review, the analysis is guided by these dimensions, considering quality metrics, computational efficiency, real-time feasibility, and the characteristics of the target execution platforms.

There are existing review studies that investigate the development of solutions for VQE [51], as well as surveys specifically focused on super-resolution models [23]. Similarly, several reviews organize research related to the use of DNNs on edge devices [6, 39]. Although previous surveys have extensively investigated video quality enhancement, video super-resolution, and deep learning for edge computing, these studies generally address these topics from broader or distinct perspectives in particular, existing reviews focus either on the development of VQE and VSR techniques, or on the broader application of deep learning in edge environments, without systematically analyzing the architectural, computational, and deployment characteristics of DNN-based VQE solutions under edge-device constraints. Therefore, this review focuses specifically on the intersection between video quality enhancement and edge computing, emphasizing the trade-offs among reconstruction quality, computational complexity, memory consumption, energy efficiency, and real-time performance.

This study is organized as follows. Section 2 describe the methodology used in this review. Section 3 formulates the problem of VQE in the context of edge devices. Section 4 provides an overview of the main advances in DNN models for VQE. Section 5 discusses the

principal model optimization techniques employed to enable real-time performance. Section 6 reviews and analyzes state-of-the-art DNN architectures for VQE targeting edge computing platforms. Section 7 summarizes the key challenges identified in the literature, and finally, Section 8 concludes the study.

2 REVIEW METHODOLOGY

This review adopts a structured literature analysis methodology to identify, categorize, and analyze DNN based VQE solutions targeting edge devices. The objective of this methodology is to ensure transparency in the selection process while maintaining flexibility to support a technical and architecture-oriented discussion rather than a strictly systematic bibliometric analysis.

The literature search was conducted using scientific digital libraries, including IEEE Xplore, ACM Digital Library and Google Scholar. The search focused on journal articles and conference papers published primarily in the last five years, reflecting recent advances in deep learning-based video enhancement.

The search queries combined keywords related to video enhancement tasks, deep learning techniques, and edge computing constraints. These terms were combined using Boolean operators (AND, OR) to construct the final search string, that can be summarized as:

```
("edge-based" OR "edge" OR "mobile" OR "embedded"
OR "on-device") AND ("video super-resolution"
OR "video quality enhancement" OR "VQE" OR
"compressed video enhancement" OR "neural
video enhancement") AND ("deep learning" OR
"CNN" OR "DNN" OR "lightweight network" OR
"efficient network") AND ("real-time" OR "resource
constrained" OR "low latency" OR "low complexity"
OR "low power")
```

These queries were adapted to the syntax of each database. The selection of studies was based on predefined Inclusion and Exclusion Criteria (IC and EC), as described below. Studies were included in the review if they met the following criteria:

- (IC-I) - Proposed DNN-based methods for VQE, including compression artifact reduction, denoising, or video super-resolution;
- (IC-II) - Explicitly targeted or evaluated performance on edge-related platforms, such as smartphones, single-board computers, embedded systems, or GPUs representative of edge clients;
- (IC-III) - Reported quantitative results in terms of visual quality metrics, computational complexity, latency, or frame rate;
- (IC-IV) - Articles written in English or Portuguese;
- (IC-V) - Articles with full text.

Studies were excluded if they:

- (EC-I) - Focused exclusively on single-image enhancement;
- (EC-II) - Lacked experimental validation;
- (EC-III) - Relied solely on server-side processing without executing the DNN on the edge device;
- (EC-IV) - Did not provide sufficient information to assess computational efficiency or real-time feasibility;

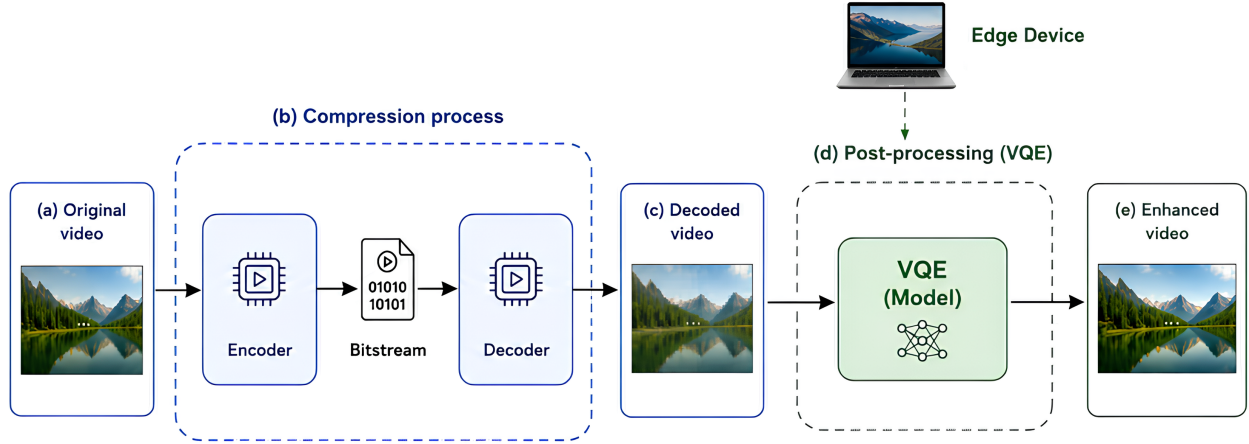


Figure 1: General pipeline of video compression and quality enhancement on edge devices. A compressed video is decoded and subsequently enhanced by a VQE model executed directly on the edge device.

(EC-V) - Conference abstracts, short papers, or publications without sufficient technical detail and experimental evidence for the analysis performed in this review.

2.1 Study Selection and Analysis

The study selection process was conducted in multiple stages to progressively refine the set of relevant works. The initial search across the selected digital libraries resulted in a total of 245 studies. In the first screening stage, titles and abstracts were reviewed to assess relevance to DNN-based VQE and edge-oriented execution. After the application of ICs and ECs described in the previous section, the number of candidate studies is reduced to 52.

In the subsequent stage, a full-text review was performed to evaluate the alignment of each study with the scope of this review, particularly regarding architectural design, real-time performance, and validation on edge-related platforms.

After the full reading of each article, a quality assessment score was computed based on a set of predefined evaluation questions and their corresponding weighted answers. Each question could receive a score of 1.0 (“Yes”), 0.5 (“Partially”), or 0.0 (“No”), and the final quality score of an article was obtained by summing the weights across all questions. Considering that the maximum possible score was 3.0, a cutoff score of 2.0 was established to determine study eligibility. Articles scoring below this threshold were excluded, as they did not sufficiently meet the minimum quality requirements defined for this review. The quality assessment was conducted according to the following questions:

- (1) Does the article present a neural network architecture for video quality enhancement?
- (2) Is the proposed architecture specifically designed for edge devices?
- (3) Does the article perform quality improvement evaluation and computational complexity analysis?

After this detailed analysis, a final set of 22 studies was selected for inclusion.

The selected works were analyzed qualitatively, with emphasis on architectural design choices, optimization strategies, validation platforms, and reported trade-offs between visual quality and computational efficiency. These works were subsequently organized into thematic categories, including system-level video delivery frameworks and optimized neural architectures, enabling a structured and comparative discussion.

3 PROBLEM DEFINITION

In VQE tasks, the central objective is to learn a mapping between a degraded video and its reference version, corresponding to the original signal prior to the degradation process. In the context of video compression, this problem can be formulated as the estimation of a function that maps a compressed video V_c to an approximation of the original uncompressed video V_{ref} , i.e.,

$$\hat{V} = f_{\theta}(V_c),$$

where $f_{\theta}(\cdot)$ denotes a deep neural network parameterized by θ , and $\hat{V}$ represents the reconstructed video with enhanced quality.

With respect to the enhancement tasks themselves, architectures designed for compression artifact reduction and denoising operate by mapping low-quality videos (degraded by video codec compression or by the artificial addition of noise) into reconstructed versions with the same spatial resolution as the original video. In this case, the problem can be expressed as

$$\hat{V} = f_{\theta}(V_{lq}),$$

This process is designed such that the network learns to suppress degradation patterns that negatively impact QoE, while simultaneously preserving relevant structural details.

In contrast, video super-resolution architectures follow a different formulation. In these approaches, the video resolution is artificially reduced prior to being fed into the neural network, typically through downsampling operations. The network is then trained to reconstruct a higher-resolution version of the video, commonly

using scaling factors of $\times 2$ or $\times 4$. Formally, this problem can be described as

$$\mathbf{V}_{lr} = \mathcal{D}(\mathbf{V}_{hr}), \quad \hat{\mathbf{V}}_{hr} = f_{\theta}(\mathbf{V}_{lr}),$$

where $\mathcal{D}(\cdot)$ denotes the spatial degradation operator, $\mathbf{V}_{lr}$ represents the low-resolution video, and $\hat{\mathbf{V}}_{hr}$ is the reconstructed output produced by the network. The objective of these architectures is to recover spatial details lost during the resolution reduction process, while maintaining temporal coherence and adequate perceptual quality.

To evaluate the effectiveness of these architectures, objective metrics based on pixel-wise error are widely employed in the literature. Among them, the Peak Signal-to-Noise Ratio (PSNR) [21] is one of the most commonly used, as it quantifies the average difference between the reconstructed video and the reference video.

Although PSNR is simple and computationally efficient, its correlation with human visual perception is limited, especially in scenarios where small structural variations can have a significant impact on the QoE. In order to better capture structural aspects of visual quality, metrics such as the Structural Similarity Index Measure (SSIM) [41] are frequently adopted. SSIM evaluates the similarity between two images by jointly considering luminance, contrast, and structural information.

Beyond these classical metrics, more recent approaches incorporate perceptual metrics that aim to better align automatic evaluation with human perception. Among them, the Learned Perceptual Image Patch Similarity (LPIPS) [52] stands out, as it measures the distance between representations extracted from pretrained deep neural networks. For video assessment, metrics such as the Video Multi-Method Assessment Fusion (VMAF) [32] combine multiple perceptual and structural indicators, producing scores that exhibit a high correlation with subjective QoE evaluations.

Objective and perceptual quality metrics are essential for evaluating the effectiveness of VQE methods. However, in the context of edge devices, the development of such architectures requires explicitly considering the trade-off between gains in visual quality and the associated computational cost. Improvements in metrics such as PSNR, SSIM, or VMAF typically lead to increases in the number of parameters, arithmetic operations, energy consumption, and memory usage, which compromises real-time execution on resource-constrained platforms. Therefore, VQE architectures for edge devices must be jointly evaluated in terms of achieved quality and computational efficiency, prioritizing solutions that maximize perceptual quality within the available processing, memory, and energy constraints.

4 VIDEO QUALITY ENHANCEMENT

Initially, the VQE problem was addressed using neural networks that process a single frame at a time. The Super Resolution Convolutional Neural Network (SRCNN) model [15], proposed for image resolution enhancement, and Artifact Reduction Convolutional Neural Network (ARCNN) [49], designed to reduce compression artifacts while preserving the original spatial resolution, are representative examples of this paradigm. Both approaches rely on shallow convolutional networks and do not consider temporal information.

Figure 2 summarizes the historical evolution of DNN-based VQE and VSR architectures. Early approaches, such as SRCNN and ARCNN, relied on single-frame convolutional processing. Later works introduced temporal fusion mechanisms to exploit redundancy across neighboring frames, as observed in MFQE. More recent architectures incorporated deformable convolutions, bidirectional temporal propagation, and attention-based mechanisms to improve alignment and reconstruction quality.

Multi Frame Quality Enhancement (MFQE) [46] was one of the first works to explicitly incorporate temporal information into the VQE process by jointly processing multiple frames. The architecture introduces a temporal alignment and fusion scheme in which a sliding window of n temporally neighboring frames is used to enhance the quality of a central frame. In this framework, the alignment stage aims to compensate for motion between adjacent frames, while the fusion stage exploits temporal redundancy to reconstruct details degraded by compression. This window-based temporal processing paradigm has become a fundamental approach in VQE and was later adopted and refined by several works, such as Spatial Temporal Deformable Fusion (STDF) [14] and Video Restoration framework with Enhanced Deformable Video Restoration (EDVR) [40]. In particular, these methods replaced the explicit motion estimation via optical flow employed in MFQE with deformable convolutions, enabling more flexible and robust spatial alignment while reducing sensitivity to motion estimation errors.

The STDF architecture has also served as a foundation for subsequent adaptations. MD-STDF extended the original framework by introducing a multi-domain strategy to address video quality enhancement across different video compression codecs [11]. Similarly, 3D-STDF extended the STDF architecture by incorporating 3D convolutions to better exploit spatio-temporal information during video quality enhancement [12]. These developments illustrate how the combination of temporal fusion and deformable convolution can provide a flexible foundation for adapting VQE architectures to different compression scenarios and computational requirements.

This type of architecture based on temporal windows and alignment-fusion mechanisms has also been widely explored to address video super-resolution problems. Works such as Video Efficient Sub-Pixel Convolutional Neural Network (VESPCN) [2] and Temporally Deformable Alignment Network (TDAN) [37] adopt similar strategies by combining multiple temporally adjacent frames to reconstruct a central frame at higher resolution, leveraging temporal redundancy to recover spatial details lost during downsampling.

In addition, techniques based on recurrent neural networks and attention mechanisms, such as Deformable Spatiotemporal Attention (DSTA) [54], achieve significant gains in objective quality metrics by more effectively exploiting temporal redundancy and adaptively selecting relevant features. Similarly, architectures grounded in transformer-based models, such as BasicVSR [5], achieve superior performance in video super-resolution and restoration tasks.

Despite the substantial gains in visual quality achieved by these techniques, their computational complexity also increased significantly, resulting in models with millions of parameters and high computational cost, which currently makes them impractical for real-time execution on edge devices with severe constraints on processing power, memory, and energy consumption, motivating the development of lightweight solutions targeting edge devices.

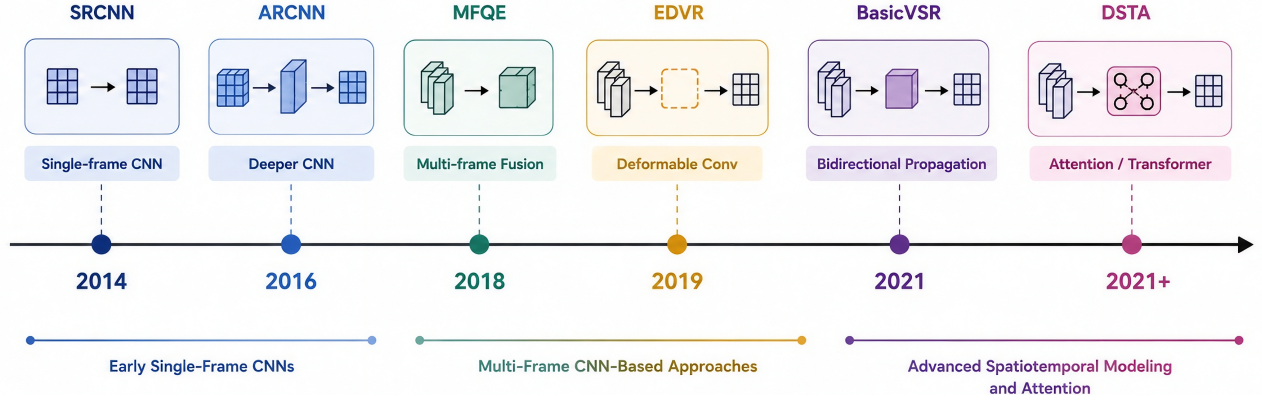


Figure 2: Historical evolution of representative DNN-based architectures for VQE, highlighting the transition from single-frame CNN approaches to advanced spatiotemporal and attention-based models.

5 OPTIMIZATION TECHNIQUES FOR REAL-TIME

Throughout this study, several key techniques that enable the design of lightweight DNN architectures for edge devices were identified.

Neural Architecture Search (NAS) [16, 24] is a technique that automates the design of neural network algorithms. Instead of relying on manual design by domain experts, NAS employs search frameworks (based on reinforcement learning, Bayesian optimization, or differentiable parameters) to automatically explore optimal arrangements of layers, channels, and operators.

Other technique is the Knowledge Distillation [36, 44] that transfers knowledge from a large and complex neural network, known as the teacher model, to a smaller and lightweight student model. The main objective is to enable efficient inference on resource-constrained devices while preserving competitive performance. Instead of learning only from raw training data, the student network is trained using the informative outputs generated by the teacher, allowing improved accuracy with reduced computational complexity.

There are also techniques specifically aimed at model compression, among which quantization and pruning are the most widely adopted. Quantization [16, 31, 36] reduces the numerical precision of weights and activations (e.g., from Float32 to INT8 or INT16). This significantly decreases memory usage and accelerates inference on mobile Neural Processing Units (NPUs) and GPUs. Pruning [31, 36], in turn, consists of removing redundant weights or filters that do not significantly contribute to image quality. It can be divided into weight pruning, which removes individual weight values within filters, and filter pruning, which discards entire filters while preserving a regular network structure.

Another model optimization strategy involves replacing computationally expensive operations with lighter alternatives that require fewer multiplications. Among the observed operations,

Depthwise Separable Convolutions (DSConv) [7, 44] are commonly used in place of standard convolutions, decomposing the convolution into two stages (depthwise and pointwise) and reducing the number of operations by up to nine times. The depthwise convolution applies a single filter independently to each input channel; for instance, if there are 32 channels, 32 separate filters are used, each operating on its corresponding channel. The pointwise convolution is a standard 1×1 convolution that combines the outputs of the previous stage across all channels to generate new feature representations.

Another class of techniques includes group convolutions and channel shuffle operations [26]. Group convolution divides the input feature maps into G distinct groups, with each group of channels processed independently by its own set of filters and concatenated at the output. Although group convolutions are computationally efficient, they restrict information flow within each group, preventing interaction across different channel groups [44]. To address this limitation, channel shuffle rearranges the output channels such that subsequent layers receive information from all previous groups.

Similarly, some studies on video super-resolution employ the PixelShuffle operation as an upsampling method [34, 38]. Unlike traditional approaches such as deconvolution, which involve computationally intensive arithmetic operations, PixelShuffle performs upsampling by rearranging data across the network dimensions. For a scaling factor F , the network produces a feature map with a large number of channels (I_c) at a low spatial resolution. PixelShuffle then reorganizes these elements to generate an output whose width and height are multiplied by F , while reducing the number of channels by a factor of F^2 .

6 ARCHITECTURES FOR EDGE DEVICES

Mechanisms such as optical flow, deformable convolutions, and transformers are computationally expensive and, although they

improve reconstruction quality, they introduce significant processing overhead. Research efforts focused on developing solutions for edge devices aim to achieve real-time performance under constrained computational budgets. Consequently, several approaches have been proposed to design lightweight models with a reduced number of parameters and lower memory requirements, enabling deployment on edge platforms. These studies can be broadly categorized into two main groups: video delivery frameworks and models optimized for smartphones.

6.1 Systems / Frameworks

Framework-based solutions are defined by their proposed execution pipelines for video delivery systems. Within this category, only studies in which the neural network model for VQE is executed on the edge device were considered.

Neural-enhanced video streaming on mobile devices (NEMO) [48] was the first framework to enable real-time super-resolution on commercial mobile devices by adopting a selective processing strategy. The system applies super resolution only to anchor points, corresponding to selected frames, and reuses the results for subsequent frames using information extracted from the video codec. This approach achieved a speed-up of 11.5 $\times$ compared to frame-by-frame VSR, reduced energy consumption by 88.6%, and delivered quality improvements ranging from 0.99 to 6.34 dB in PSNR. Following a similar idea aimed at avoiding redundant computations through the reuse of previously processed spatial or temporal information, Real-Time Neural-Enhanced Mobile Video Streaming Framework (REM) [4] employs a Macroblock-aware Lookup Table (LUT). The system stores super resolution results and reuses them for redundant macroblocks across frames, increasing throughput by up to 10.2 $\times$ and reducing energy consumption by 58.6% compared to the state of the art.

The study by [27] presented an optimized pipeline for thermal cameras on edge devices such as the Jetson Nano, exploiting a dual fine-tuning scheme to enable real-time processing even with limited thermal datasets. This method achieved processing speeds of 29, 36, and 45 FPS for scaling factors of 2 $\times$, 3 $\times$, and 4 $\times$, respectively, with a PSNR of 32.9 dB at the 2 $\times$ scale.

Moving beyond purely local solutions, Video Streaming system with Edge-Client collaborative Super-resolution (VSECS) [25] introduces a collaborative super-resolution system between the edge server and the mobile client, in which the workload partitioning decision is guided by an A3C-based reinforcement learning algorithm. In this setup, a large-scale super-resolution model runs on the edge server, while a lightweight model operates on the mobile device, allowing both to jointly reconstruct high-resolution frames. As a result, the system demonstrated improvements of up to 10% in QoE compared to solutions that rely exclusively on either the edge or the client.

In a complementary line of systemic optimization for video streaming, Bidirectionally optimized Super-Resolution (BiSR) [50] was developed as a video delivery system with bidirectional optimization. In this approach, the server performs intelligent downscaling of key frames (I-frames) using a DNN, while the mobile client executes a lightweight super-resolution network optimized for half-precision (float16) to restore these frames. The system achieved

a 26% bitrate reduction compared to traditional H.264 and a 3.7% improvement in SSIM relative to the previous state of the art.

Some studies investigate what content should be processed by applying neural networks only to visually important regions or by adapting the algorithm to image complexity. Edge-based video delivery with salience-aware super-resolution (EVASR) [22] is an edge-based framework that employs saliency-aware super-resolution. The edge server processes only image blocks with high visual relevance, while simple interpolation is applied to the remaining regions. The framework supports real-time 1080p processing and significantly improves the average VMAF score to 94.47. Complementarily the system of [16], designed for IoT scenarios with severe bandwidth constraints, adopts a hardware-aware NAS framework to design lightweight super-resolution models that explicitly account for the real latency of the edge device, achieving up to a 1.8 $\times$ acceleration for the complete system.

Table 1 presents information regarding the target platforms of the analyzed systems, the datasets used during the evaluation phase, and the types of DNNs employed. It can be observed that most studies focus on commercial smartphones from different manufacturers, reflecting an interest in solutions directly applicable to devices widely adopted by end users. In addition, single-board computer platforms, such as the NVIDIA Jetson Nano, also appear frequently due to their GPU acceleration capabilities and widespread use in edge computing scenarios.

With respect to the datasets, there is a clear predominance of videos sourced from YouTube, often organized into popular content categories. This indicates a concern with evaluating the systems on data representative of real-world video consumption scenarios on mobile devices. Additionally, some studies rely on standardized datasets such as Set5, Set14, BSD100, and Urban100, enabling more controlled comparisons with the state of the art.

Regarding the adopted DNN architectures, a clear predominance of lightweight CNN-based models can be observed, with emphasis on variants of FSRCNN and ESPCN, as well as adaptations of architectures such as SR-ResNet and RFDN. In general, these models avoid computationally expensive temporal modeling mechanisms, such as optical flow, deformable convolutions, or explicit temporal attention.

6.2 Optimized Models

Optimized model solutions encompass the development of new DNN building blocks and architectures with a strong focus on model efficiency.

Efficient Video Restoration Network (EVRNet) [28] is an efficient network designed for multiple restoration tasks (deblocking, denoising, and super-resolution) using lightweight encoder-decoder-based alignment and fusion modules. Although it achieves lower objective quality compared to heavier models, EVRNet has 260 $\times$ fewer parameters and 958 $\times$ fewer MACs than EDVR [40] for $\times 4$ super-resolution. It achieved 36.65 dB PSNR for the deblocking task on the Vimeo-90K dataset. Similarly, the Lightweight Dense residual Network (LiDeR) architecture [3] was developed as a lightweight dense residual network tailored for mobile GPUs, enabling parallel processing of up to 10 frames. It achieved 139 FPS for $\times 4$ upscaling on the Snapdragon 888 GPU, being 267% faster than the

System/Framework	Device	Test Dataset	DNN
NEMO [48]	Smartphones (Xiaomi Mi9 – Snapdragon 855)	30 YouTube videos (10 categories)	Modified SR-ResNet (uses deconvolution instead of sub-pixel).
VSECS [25]	Server (PC) + Client (Jetson TX2)	YouTube (10 popular categories)	SRResNet (Server) and Fast Super-Resolution Convolutional Neural Network (FSRCNN) (Client).
[27]	NVIDIA Jetson Nano	DIV2K + Custom thermal dataset	Efficient Sub-pixel Convolutional Neural Network (ESPCN), FSRCNN, and Laplacian Pyramid Super-Resolution Network (LapSRN).
BiSR [50]	Smartphones (Redmi K30, Galaxy A90)	YouTube 4K (Interview, Game, Chat)	Residual Feature Distillation Network (RFDN) with a novel channel attention mechanism; dedicated downscaling DNN.
PASR [16]	LicheePi 4A (RISC-V TH1520)	Set5, Set14, BSD100, Urban100	Models obtained via hardware-aware NAS.
REM [4]	Xiaomi 13, Honor V30 Pro, Jetson Nano	YouTube (6 categories)	Macrobloc-aware SR module.
EVASR [22]	Google Pixel 3, Jetson Nano	UVG (4K), YouTube	ESPCN.

Table 1: VQE Systems / Frameworks for edge devices.

ESPCN model. Efficient Video Super Resolution Network (EVSNet) [24] is an architecture obtained via fine-grained neural architecture search (FGNAS) with the goal of minimizing FLOPs and latency on mobile devices. The model achieved second place in the Mobile AI 2021 challenge, reaching a PSNR of 27.85 dB with a latency of only 11.3 ms per frame on the Snapdragon 865 SoC.

Several studies have proposed novel DNN building blocks that contribute to the final performance of complete architectures. One such example is [53], which introduces the Edge-oriented Convolution Block (ECB). This block re-parameterizes multiple training-time paths (including convolutions and first-/second-order derivatives) into a single 3×3 convolution at inference time. This approach achieved PSNR values of up to 31.04 dB and runs at 0.033 s per frame on a Snapdragon 865. Another study that explored new block designs is Efficient Lightweight Network for Video Super-Resolution (ELNVSr) [26], which introduces the Bidirectional Alignment Module (BAM) for implicit bidirectional alignment between adjacent frames and a Multi-scale Pyramidal Block (MPB) for reconstruction. The MPB employs depthwise separable convolutions to extract features at multiple spatial scales in a pyramidal manner. The model achieved 46.5 FPS on the Vid4 dataset, with an average PSNR of 26.13 dB.

SR-UNet [38] is an adaptation of the UNet architecture with a reduced size (limited to 64 filters) and trained using perceptual GAN-based objectives. Although the model does not report classical objective metrics such as PSNR or SSIM, it achieved a VMAF score of 84.30 on the Derf's Collection dataset while running at 46.1 FPS on a GTX 1080Ti, being significantly faster than traditional residual-based models. Similarly, UNetDeblur [31] follows a UNet-based design and integrates Mobile Vision Transformer (ViT) and

Convolutional Gated Recurrent Units (ConvGRUs) into the architecture for real-time mild motion deblurring. The model achieved processing speeds between 30 and 35 FPS on the Qualcomm Snapdragon 8 Gen 2, while maintaining a PSNR of approximately 30 dB on real-world videos.

The study in [44] introduced Lightweight Shuffle Video Super-Resolution (LSVSR), which employs depthwise separable group convolutions to drastically reduce computational cost. To mitigate the lack of inter-group communication, the network applies a Channel Shuffle module that rearranges channels to enable information flow. The model achieved a PSNR of 25.86 dB on the Vid4 dataset, with an inference time of 29 ms for $\times 3$ upscaling on an NVIDIA 980 Ti GPU. ELSR [45] is a serial network composed of only six layers, designed for extremely low power consumption (0.09 W) and high processing speed. The model won the MAI & AIM 2022 challenge and achieved a power consumption of only 0.09 W on a Dimensity 9000 chip, while maintaining a PSNR of 27.34 dB on the REDS dataset.

Some approaches focus on specific video codecs, such as FastCNN [17], which targets HEVC and proposes the EAM module to progressively estimate deformable offsets at low resolutions, thereby reducing computational cost. The resulting model reduced inference time by up to 69% and improved PSNR by up to 0.11 dB compared to the state of the art at the time. Following a similar idea, Real Time Super Resolution (RTSR) [19] is an optimized model for AV1-compressed videos based on dual-teacher knowledge distillation. The model achieved the best trade-off between complexity and performance (PSNR, SSIM, and VMAF) in the AIM 2024 Video SR Challenge. With a more direct integration with the video codec, the Codec Information-assisted VSR (CILVSR) architecture [20]

Architecture	Device	Resolution (Input → Output)	Parameters	FLOPs / MACs	FPS	Performance (Metric)
LSVSR [44]	NVIDIA 980 Ti	540p → 2160p (4K)	40 K	1,22e10 operations	~34,5	25,86 dB (PSNR)
EVSNet [24]	Snapdragon 865	180p → 720p (x4)	N/A	N/A	~88,5	27,85 dB (PSNR)
EVNet [40]	iPhone 11	180p → 720p (x4)	79,5 K	10,39 G MACs	20,4	35,98 dB (PSNR)
ECBSR [53]	Snapdragon 865	270p → 1080p (x4)	11,9 K	0,69 G FLOPs	30,3	31,04 dB (PSNR)
SR-UNet [38]	GTX 1080Ti	540p → 1080p	741 K	N/A	46,1	84,30 (VMAF)
ELSR [45]	MediaTek Dimensity 9000	180p → 720p (x4)	N/A	N/A	30	27,34 dB (PSNR)
LiDeR [3]	Xiaomi Mi 11	180p → 720p (x4)	N/A	N/A	139	27,51 dB (PSNR)
CILVSR [20]	RTX 6000	135p → 540p (x4)	121 K	2,8 G FLOPs	25	23,89 dB (PSNR)
FastCNN [17]	RTX 3090	1080p	425 K	75,8 G FLOPs	17,6	+0,85 dB (Δ PSNR)
Fast-MFQE [7]	GTX 1080Ti	1080p	243 K	N/A	25,7	+0,67 dB (Δ PSNR)
FDAN [47]	RTX 2080Ti	180p → 720p (x4)	3,9 M	292,0 G FLOPs	29,4	30,71 dB (PSNR)
ELNVSR [26]	GPU	135p → 540p (x4)	N/A	N/A	46,5	26,13 dB (PSNR)
STDF [13]	GTX 1080Ti	240p	480 K	72,2 G FLOPs	45	32,53 dB (PSNR)
RTSR [19]	RTX 3090	360p → 1080p (x3)	63 K	1,16 K/pixel	~1250	34,46 dB (PSNR)
UNetDeblur [31]	Snapdragon 8 Gen 2	N/A	N/A	N/A	35	29,1 dB (PSNR)

Table 2: Reported computational and performance characteristics of representative edge-oriented VQE architectures.

exploits information already available at the video decoder, such as motion vectors and macroblock types, to guide the super-resolution process, eliminating the need for computationally expensive motion estimation modules.

Despite the high computational cost associated with optical flow, some studies retain this mechanism for feature extraction while still achieving real-time performance under certain conditions. The Flow-guided deformable alignment network (FDAN) network [47] combines a lightweight optical flow network (FastFlowNet) with a deformable attention module (FDAP) to enable fast online super-resolution. Optical flow provides guidance for deformable attention to more efficiently locate corresponding pixels in neighboring frames. The model achieved 29.4 FPS on the REDS4 dataset for $\times 4$ upscaling, being three times faster than EDVR-M while delivering superior quality.

Following an approach of updating existing architectures to make them more lightweight, Fast-MFQE [7] was the first lightweight method for multi-frame quality enhancement using fusion attention and the IPPB preprocessing module. The resulting model achieved real-time performance, reaching more than 25 FPS at 1080p resolution with an average PSNR improvement of 19.6%. Similarly, the study in [13] proposes a modification of the STDF architecture to make it more suitable for embedded devices. The work introduces spatial-temporal deformable convolution (STDC) to replace the computationally expensive optical flow in embedded systems. The resulting model increased the average PSNR gain from 0.52 to 0.81 dB (a 56% improvement) while achieving an 80% speedup over the previous state of the art.

Table 2 presents a comparison of the analyzed architectures, considering aspects related to model computational complexity and

the platforms used for validation. It can be observed that a significant portion of the studies conducted experiments on commercial GPUs. Although such devices are not typically classified as highly constrained edge platforms, their widespread adoption in end-user clients and edge computing scenarios justifies their inclusion in this review. It is important to emphasize that the results presented were directly extracted from the original studies, without reimplementation or experimental re-evaluation. Since the analyzed works employ different datasets, codecs, compression settings, scaling factors, hardware platforms, and evaluation protocols, the reported performance values should be interpreted as indicative references of the quality–efficiency trade-off rather than as strictly standardized comparisons. These values should not be interpreted as a direct ranking of the evaluated architectures.

The results suggest that most architectures achieving high frame rates do so by processing relatively low input resolutions, such as 135p, 180p, or 270p, followed by reconstruction to higher resolutions. Methods that operate directly on higher input resolutions, such as 540p or 1080p, tend to exhibit significantly lower FPS, highlighting that reducing input resolution remains a fundamental strategy for enabling real-time execution. It is also observed that architectures with a reduced number of parameters, such as ECBSR, EVSRNet, and LSVSR, can achieve frame rates compatible with real-time processing, particularly when combined with lower input resolutions.

Nevertheless, the results suggest that models performing upsampling from extremely low input resolutions, such as 135p or 180p, tend to achieve significantly lower PSNR values when compared to architectures operating on intermediate input resolutions. This trend can be observed, for instance, in methods such as CILVSR and ELNVSR, which, despite achieving real-time performance, report PSNR values in the range of approximately 23–26 dB. In contrast, architectures that operate on higher input resolutions, such as 270p or 540p, are able to achieve higher PSNR values even with relatively compact models.

Figure 3 reinforces the existence of a strong trade-off between reconstruction quality and real-time performance in edge-oriented VQE architectures. The horizontal axis adopts a logarithmic scale due to the large variation in FPS values among the analyzed architectures, allowing both low-latency and extremely high-throughput models to be visualized simultaneously without compressing the distribution of lower FPS results. Models operating on extremely low input resolutions generally achieve higher FPS because the number of pixels processed by the network is substantially reduced, directly lowering the computational cost of convolutional operations and memory access. This behavior can be observed in architectures such as EVSRNet and LiDeR, which process low-resolution inputs and achieve high frame rates while maintaining relatively compact network structures.

However, the reduction of input resolution also limits the amount of spatial information available for reconstruction. When the input is excessively degraded, important high-frequency details are permanently lost before the enhancement stage, forcing the network to estimate missing structures from limited contextual information. As a consequence, methods such as CILVSR and ELNVSR, despite achieving real-time execution, present lower PSNR values compared to architectures operating on intermediate resolutions.

In contrast, architectures processing higher input resolutions, such as ECBSR, STDF, and EVRNet, preserve a larger amount of spatial information prior to enhancement, enabling superior reconstruction quality even when using lightweight architectures. Nevertheless, this improvement comes at the cost of increased computational complexity and reduced frame rate, since the number of operations grows proportionally with the spatial dimensions of the processed frames.

The results also suggest that model efficiency is not determined exclusively by the number of parameters. For instance, FDAN presents considerably higher computational complexity due to the use of optical flow and deformable attention mechanisms, resulting in a substantially larger model despite achieving competitive PSNR values. Conversely, compact architectures such as ECBSR demonstrate that carefully designed lightweight convolutional blocks can achieve favorable trade-offs between quality and efficiency without relying on computationally expensive temporal modeling strategies.

Overall, the analyzed architectures indicate that real-time VQE on edge devices is fundamentally constrained by the balance between input resolution, reconstruction fidelity, and computational complexity. While aggressive downsampling enables real-time processing, it also reduces the upper bound of achievable perceptual quality, highlighting the importance of developing architectures capable of maximizing reconstruction quality under strict computational budgets.

Table 3 summarizes the main optimization strategies adopted by edge-oriented VQE architectures. The analysis of the optimization techniques table reveals that enabling VQE on edge devices does not rely on a single strategy, but rather on a combination of approaches adapted to the target hardware platform. Quantization emerges as the most widely adopted technique, being employed in architectures such as PASR and LDRCN, and establishing itself as a fundamental method for reducing resource consumption and accelerating inference through lower-precision representations such as INT8 and FP16.

In addition, architectural efficiency is frequently achieved by replacing computationally expensive operators with DSConv and Group Convolutions, as observed in models such as LSVSR and Fast-MFQE, enabling frame rates above 25 FPS while maintaining low parameter counts.

7 MAIN CHALLENGES

The development of VQE solutions for edge devices faces critical challenges that lie at the intersection of high algorithmic complexity and the severe resource constraints of mobile hardware.

(a) Hardware and Energy Constraints: DNN inference on mobile devices is highly energy demanding, potentially draining smartphone batteries within a few hours or consuming up to 14 times more energy than standard video decoding [50]. Moreover, intensive use of processing units (GPU/NPU) generates excessive heat, causing devices to reach temperatures that are uncomfortable for users (above 40–45 °C) [48]. Another frequently reported issue is the heavy reliance on external memory access (DRAM), which is both slow and energy intensive [33, 55]. This is mainly due to the limited on-chip memory capacity of mobile devices, which

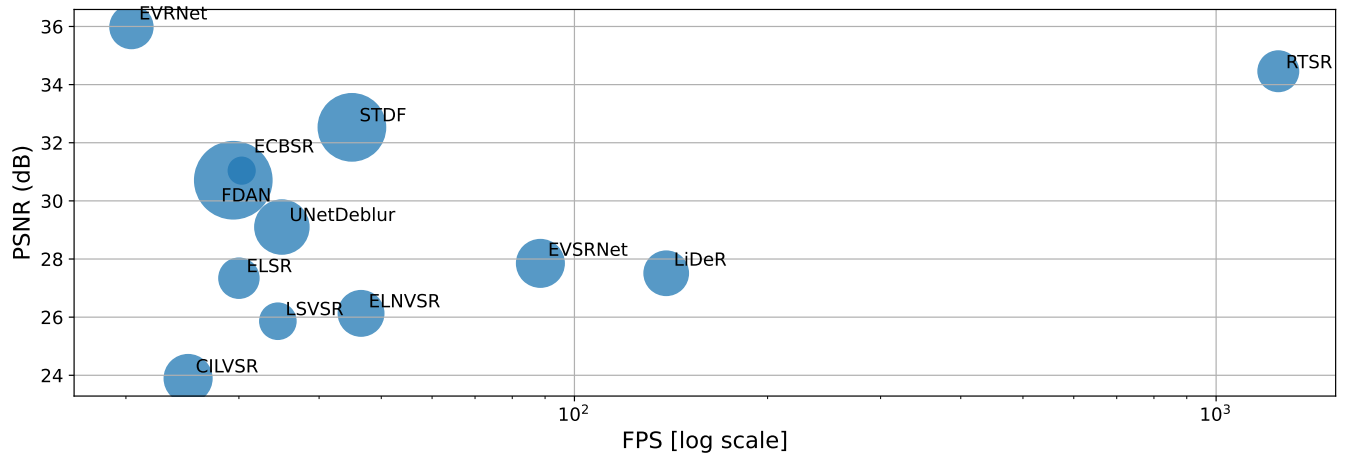


Figure 3: Trade-off between reconstruction quality (PSNR) and real-time performance (FPS) for representative edge-oriented VQE architectures. Bubble size indicates the approximate number of model parameters. The horizontal axis uses logarithmic scale.

Architecture	NAS	Quantization	Pruning	Knowledge Distillation	DSConv/ GConv	Shuffle (Channel/Pixel)
PASR [16]	v	v	v			
SR-UNet [38]		v	v			
ELVSR [33]						v
EVSRNet [24]	v					v
LSVSR [44]					v	v
Fast-MFQE [7]					v	
NEMO [48]		v				
UNetDeblur [31]		v	v			
RTSR [19]				v		v
ECBSR [53]		v				

Table 3: Mapping of Optimization Techniques per Study.

prevents storing high-resolution video feature maps locally, thereby hindering real-time processing.

(b) Algorithmic Complexity and Real-Time Constraints: Many models achieve high visual quality but fail to operate at the frame rates required for video streaming [17, 28]. High-fidelity video restoration requires exploiting correlations among neighboring frames; however, techniques such as optical flow or deformable convolutions are computationally expensive and difficult to deploy on low-power architectures without introducing significant latency. Adapting such solutions for real-time execution requires carefully investigating the trade-off between output quality and resolution of the neural network and the computational capabilities of the target hardware.

(c) Model optimization: Optimizing an already trained model for efficient deployment presents several challenges, particularly when applying post-training techniques. Reducing numerical precision requires carefully balancing bit-width reduction against acceptable accuracy loss, as aggressive quantization can lead to severe performance degradation [55]. Model behavior is often highly sensitive to parameters with large dynamic ranges, making quantization susceptible to rounding errors unless activation scaling is carefully calibrated. Similarly, parameter pruning introduces structural challenges, as removing individual weights may result in irregular sparsity patterns that are not efficiently supported by standard hardware accelerators. Furthermore, common pruning heuristics,

such as assuming that parameters with smaller norms are less important, do not consistently hold, and the removal of seemingly minor components can significantly degrade reconstruction quality [8]. Finally, pruning strategies are frequently tightly coupled to the original network architecture, limiting their generalizability across different models and tasks and complicating their adoption as a universal optimization approach [42].

(d) Platform Heterogeneity: The wide range of chips with varying computational capabilities makes it difficult to design a single solution that performs well across both entry-level and flagship smartphones. Furthermore, this platform heterogeneity complicates result interpretation, as reported gains may largely stem from the use of more powerful hardware rather than intrinsic architectural improvements. The lack of standardized evaluation protocols limits result reproducibility and prevents fair comparisons of the quality–efficiency trade-off. Consequently, the standardization of validation platforms emerges as a fundamental challenge for consolidating the state of the art in edge-oriented VQE/VSR.

(e) Lack of Standardized Evaluation Protocols: The lack of standardized evaluation protocols remains a major limitation for comparing edge-oriented VQE architectures. Existing studies employ different datasets, input resolutions, codecs, compression settings, hardware platforms, inference implementations, and measurement procedures. As a result, differences in reported FPS, latency, energy consumption, and reconstruction quality cannot always be attributed exclusively to architectural improvements. Establishing standardized evaluation protocols, including common datasets, resolutions, hardware platforms, quality metrics, and efficiency measurements, would enable more reliable comparisons and improve the reproducibility of future research.

8 CONCLUSION

This review analyzed the state of the art in Deep Neural Network-based Video Quality Enhancement targeting edge devices, with a particular focus on real-time execution under strict resource constraints. The surveyed works demonstrate that while advanced mechanisms such as optical flow, deformable convolutions, recurrent models, and transformers can significantly improve visual quality, their computational and energy costs often render them impractical for deployment on resource-limited platforms. As a result, most edge-oriented solutions favor lightweight CNN-based architectures, selective processing strategies, and reduced input resolutions to achieve feasible frame rates.

The analysis reveals two dominant research directions: system-level frameworks that optimize video delivery pipelines by reusing codec information or distributing computation across edge and client devices, and model-level approaches that introduce efficient architectural blocks, compact networks, and hardware-aware optimizations. Despite notable progress, several challenges remain unresolved, including the lack of standardized validation platforms, the difficulty of fairly comparing models across heterogeneous hardware, and the inherent trade-off between perceptual quality and computational efficiency. Moreover, results indicate that architectures operating on extremely low input resolutions often struggle to achieve high objective quality metrics, highlighting a fundamental limitation of aggressive resolution reduction strategies.

As future work, this study will conduct a more in-depth investigation of the main architectures identified in the literature, focusing on a comprehensive and fair analysis of their performance and energy consumption under standardized conditions, using the same equipment and experimental setup. The goal is to preserve competitive visual quality while significantly reducing computational cost, memory footprint, and energy consumption, enabling robust real-time VQE deployment on widely adopted edge platforms.

9 ACKNOWLEDGEMENTS

This study was financed in part by the *Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001*, Foundation for Research Support of the State of Rio Grande do Sul (FAPERGS), and National Council for Scientific and Technological Development (CNPq).

10 ETHICAL CONSIDERATIONS

This study does not involve human participants or the collection of personal data. The experiments use publicly available video datasets, and no personally identifiable information was collected or processed. Therefore, ethical approval and informed consent were not required.

During the preparation of this work, the authors used ChatGPT (OpenAI) to support linguistic revision, improve textual clarity, and assist in creating some illustrative figures. All generated content was reviewed and validated by the authors, who remain responsible for the final content of the manuscript.

REFERENCES

- [1] Benjamin Bross, Ye-Kui Wang, Yan Ye, Shan Liu, Jianle Chen, Gary J Sullivan, and Jens-Rainer Ohm. 2021. Overview of the versatile video coding (VVC) standard and its applications. *IEEE Transactions on Circuits and Systems for Video Technology* 31, 10 (2021), 3736–3764.
- [2] Jose Caballero, Christian Ledig, Andrew Aitken, Alejandro Acosta, Johannes Totz, Zehan Wang, and Wenzhe Shi. 2017. Real-time video super-resolution with spatio-temporal networks and motion compensation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4778–4787.
- [3] Ekrem Çetinkaya, Minh Nguyen, and Christian Timmerer. 2022. LiDeR: Lightweight dense residual network for video super-resolution on mobile devices. In *2022 IEEE 14th Image, Video, and Multidimensional Signal Processing Workshop (IVMSP)*. IEEE, 1–5.
- [4] Baili Chai, Di Wu, Jinyu Chen, Mengyu Yang, Zelong Wang, and Miao Hu. 2024. REM: Enabling Real-Time Neural-Enhanced Video Streaming on Mobile Devices Using Macrobloc-Aware Lookup Table. *IEEE Transactions on Mobile Computing* (2024).
- [5] Kelvin CK Chan, Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. 2021. Basicvsr: The search for essential components in video super-resolution and beyond. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 4947–4956.
- [6] Jiasi Chen and Xukan Ran. 2019. Deep learning with edge computing: A review. *Proc. IEEE* 107, 8 (2019), 1655–1674.
- [7] Kemi Chen, Jing Chen, Huanqiang Zeng, and Xueyuan Shen. 2023. Fast-MFQE: A fast approach for multi-frame quality enhancement on compressed video. *Sensors* 23, 16 (2023), 7227.
- [8] Lvcheng Chen, Pingyang Liu, and Li Zhang. 2021. A Light-Weight Compressed Video Processing Method on Embedded Platforms for IIoT. In *2021 International Conference on Intelligent Technology and Embedded Systems (ICITES)*. IEEE, 8–13.
- [9] CISCO. 2020. Cisco Annual Internet Report (2018–2023) White Paper. Available online: <https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html>. Accessed: 2026-01-07.
- [10] Michele Claus and Jan Van Gemert. 2019. Videnn: Deep blind video denoising. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*. 0–0.
- [11] Garibaldi da Silveira Júnior, Gilberto Kreisler, Bruno Zatt, Daniel Palomino, and Guilherme Correa. 2024. Multi-Domain Spatio-Temporal Deformable Fusion

- model for video quality enhancement. In *Brazilian Symposium on Multimedia and the Web (WebMedia)*. SBC, 223–230.
- [12] Garibaldi da Silveira Júnior, Bruno Zatt, Daniel Palomino, and Guilherme Correa. 2025. 4D-STDF: Compressed Video Quality Enhancement with 3D Spatio-Temporal Fusion and Deformable Convolution. In *Brazilian Symposium on Multimedia and the Web (WebMedia)*. SBC, 491–495.
 - [13] Jianing Deng, Shunjie Dong, Lvcheng Chen, Jingtong Hu, and Cheng Zhuo. 2024. StdF: Spatio-temporal deformable fusion for video quality enhancement on embedded platforms. *ACM Transactions on Embedded Computing Systems* 23, 2 (2024), 1–25.
 - [14] Jianing Deng, Li Wang, Shiliang Pu, and Cheng Zhuo. 2020. Spatio-temporal deformable convolution for compressed video quality enhancement. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 34. 10696–10703.
 - [15] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. 2014. Learning a Deep Convolutional Network for Image Super-Resolution. In *European Conference on Computer Vision (ECCV)*. 184–199. doi:10.1007/978-3-319-10593-2_13
 - [16] Xuyang Duan, Zikang Zhou, Wu Qiu, Zhiyi Shu, Zheng Tang, and Jun Han. 2024. A Hardware-Aware and Content-Adaptive Super Resolution System for Bandwidth-Limited IoT Video Delivery. In *Proceedings of the 14th International Conference on the Internet of Things*. 19–27.
 - [17] Zhijie Huang, Jun Sun, and Xiaopeng Guo. 2023. FastCNN: Towards fast and accurate spatiotemporal network for HEVC compressed video enhancement. *ACM Transactions on Multimedia Computing, Communications and Applications* 19, 3 (2023), 1–22.
 - [18] Andrey Ignatov, Andres Romero, Heewon Kim, and Radu Timofte. 2021. Real-time video super-resolution on smartphones with deep learning, mobile ai 2021 challenge: Report. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2535–2544.
 - [19] Yuxuan Jiang, Jakub Nawala, Chen Feng, Fan Zhang, Xiaoqing Zhu, Joel Sole, and David Bull. 2025. RTSR: A real-time super-resolution model for av1 compressed content. In *2025 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 1–5.
 - [20] Ilhwan Kwon, Jun Li, and Mukesh Prasad. 2023. Lightweight video super-resolution for compressed video. *Electronics* 12, 3 (2023), 660.
 - [21] Chulhee Lee, Seungdeuk Cho, Jihwan Choe, Taeuk Jeong, Wonsuk Ahn, and Eunjae Lee. 2006. Objective video quality assessment. *Optical engineering* 45, 1 (2006), 017004.
 - [22] Na Li, Zichen Zhu, Sheng Wei, and Yao Liu. 2025. EVASR: Edge-Based Salience-Aware Super-Resolution for Enhanced Video Quality and Power Efficiency. *ACM Transactions on Multimedia Computing, Communications and Applications* (2025).
 - [23] Hongying Liu, Zubo Ruan, Peng Zhao, Chao Dong, Fanhua Shang, Yuanyuan Liu, Linlin Yang, and Radu Timofte. 2022. Video super-resolution based on deep learning: a comprehensive survey. *Artificial Intelligence Review* 55, 8 (2022), 5981–6035.
 - [24] Shaoli Liu, Chengjian Zheng, Kaidi Lu, Si Gao, Ning Wang, Bofei Wang, Diankai Zhang, Xiaofeng Zhang, and Tianyu Xu. 2021. Evsrnet: Efficient video super-resolution with neural architecture search. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2480–2485.
 - [25] Xilai Liu, Zhihui Ke, Xiaobo Zhou, Tie Qiu, and Keqiu Li. 2022. QoE-oriented adaptive video streaming with edge-client collaborative super-resolution. In *GLOBECOM 2022-2022 IEEE Global Communications Conference*. IEEE, 6158–6163.
 - [26] Laigan Luo, Benshun Yi, Zhongyuan Wang, Peng Yi, and Zheng He. 2024. Efficient lightweight network for video super-resolution. *Neural Computing and Applications* 36, 2 (2024), 883–896.
 - [27] Prayushi Mathur, Ashish Kumar Singh, Syed Azeemuddin, Jayram Adoni, and Prasad Adireddy. 2021. A real-time super-resolution for surveillance thermal cameras using optimized pipeline on embedded edge device. In *2021 17th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE, 1–7.
 - [28] Sachin Mehta, Amit Kumar, Fitsum Reda, Varun Nasery, Vikram Mulukutla, Rakesh Ranjan, and Vikas Chandra. 2021. Evrnet: Efficient video restoration on edge devices. In *Proceedings of the 29th ACM international conference on multimedia*. 983–992.
 - [29] Netflix Tech Blog. 2022. All of Netflix's HDR video streaming is now dynamically optimized. Available online: <https://netflixtechblog.com/all-of-netflixs-hdr-video-streaming-is-now-dynamically-optimized-e9e0cb15f2ba>. Accessed: 2026-01-07.
 - [30] Seonghoon Park, Minchan Kim, Hyejin Park, Jeho Lee, Jiwon Kim, and Hojung Cha. 2025. EOS: Energy-Optimized Super-Resolution on Mobile Devices for Live 360-Degree Videos. In *Proceedings of the 31st Annual International Conference on Mobile Computing and Networking*. 878–893.
 - [31] Arti Ranjan and M Ravinder. 2025. UNetDeblur: Optimized Lightweight and Efficient Motion Deblurring for Mobile Platforms in Real-Time Scenarios. *Circuits, Systems, and Signal Processing* (2025), 1–34.
 - [32] Reza Rassool. 2017. VMAF reproducibility: Validating a perceptual practical video quality metric. In *Proceedings of 2017 IEEE international symposium on broadband multimedia systems and broadcasting (BMSB)*. IEEE, Piscataway, NJ, USA, 1–2.
 - [33] Amirhossein Sadough, Mahyar Shahsavari, Mark Wijnvliet, and Marcel van Gerven. 2025. ELVSR: An efficient and lightweight convolutional neural network for FPGA-Based FHD to UHD video super-resolution. *Array* (2025), 100542.
 - [34] Amirhossein Sadough, Mahyar Shahsavari, Mark Wijnvliet, and Marcel AJ van Gerven. 2024. ELVSR: An Efficient and Lightweight Convolutional Neural Network for FPGA-Based FHD to UHD Video Super-Resolution. Available at SSRN 5150107 (2024).
 - [35] I Sandvine. 2023. Global internet phenomena report. Available online: <https://www.applogincnetworks.com/phenomena>. Accessed: 2026-01-07.
 - [36] Sonia Victor Soans, T Prabakaran, and S Vamshi Krushna. 2025. AI-DRIVEN REAL-TIME VIDEO ENHANCEMENT USING SUPER-RESOLUTION AND OPTICAL FLOW TECHNIQUE. *ICTACT Journal on Image & Video Processing* 15, 3 (2025).
 - [37] Yapeng Tian, Yulun Zhang, Yun Fu, and Chenliang Xu. 2020. Tdan: Temporally-deformable alignment network for video super-resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 3360–3369.
 - [38] Federico Vaccaro, Marco Bertini, Tiberio Uricchio, and Alberto Del Bimbo. 2021. Fast video visual quality and resolution improvement using SR-UNet. In *Proceedings of the 29th ACM International Conference on Multimedia*. 1221–1229.
 - [39] Fangxin Wang, Miao Zhang, Xiangxiang Wang, Xiaoqiang Ma, and Jiangchuan Liu. 2020. Deep learning for edge computing applications: A state-of-the-art survey. *IEEE Access* 8 (2020), 58322–58336.
 - [40] Xintao Wang, Kelvin CK Chan, Ke Yu, Chao Dong, and Chen Change Loy. 2019. Edvr: Video restoration with enhanced deformable convolutional networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*. 0–0.
 - [41] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing* 13, 4 (2004), 600–612.
 - [42] Xinyi Wu, Santiago López-Tapia, Xijun Wang, Rafael Molina, and Aggelos K Katsaggelos. 2024. Real-time lightweight video super-resolution with RRED-based perceptual constraint. *IEEE Transactions on Circuits and Systems for Video Technology* 34, 10 (2024), 10310–10325.
 - [43] Yushu Wu, Yifan Gong, Pu Zhao, Yanyu Li, Zheng Zhan, Wei Niu, Hao Tang, Minghai Qin, Bin Ren, and Yanzhi Wang. 2022. Compiler-aware neural architecture search for on-mobile real-time super-resolution. In *European Conference on Computer Vision*. Springer, 92–111.
 - [44] Zhijiao Xiao, Zhikai Zhang, Kwok-Wai Hung, and Simon Lui. 2021. Real-time video super-resolution using lightweight depthwise separable group convolutions with channel shuffling. *Journal of Visual Communication and Image Representation* 75 (2021), 103038.
 - [45] Tianyu Xu, Zhuang Jia, Yijian Zhang, Long Bao, and Heng Sun. 2022. Elsr: Extreme low-power super resolution network for mobile devices. *arXiv preprint arXiv:2208.14600* (2022).
 - [46] Ren Yang, Mai Xu, Zulin Wang, and Tianyi Li. 2018. Multi-frame quality enhancement for compressed video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 6664–6673.
 - [47] Xi Yang, Xindong Zhang, and Lei Zhang. 2023. Flow-guided deformable attention network for fast online video super-resolution. In *2023 IEEE International Conference on Image Processing (ICIP)*. IEEE, 390–394.
 - [48] Hyunho Yeo, Chan Ju Chong, Youngmok Jung, Juncheol Ye, and Dongsu Han. 2020. Nemo: enabling neural-enhanced video streaming on commodity mobile devices. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*. 1–14.
 - [49] Ke Yu, Chao Dong, Chen Change Loy, and Xiaoou Tang. 2016. Deep convolution networks for compression artifacts reduction. *arXiv preprint arXiv:1608.02778* (2016).
 - [50] Qian Yu, Qing Li, Rui He, Gareth Tyson, Wanxin Shi, Jianhui Lv, Zhenhui Yuan, Peng Zhang, Yulong Lan, and Zhicheng Li. 2023. BiSR: Bidirectionally optimized super-resolution for mobile video streaming. In *Proceedings of the ACM Web Conference 2023*. 3121–3131.
 - [51] Jian Yue, Mao Ye, Luping Ji, Hongwei Guo, and Ce Zhu. 2025. A Survey of Deep-Learning-Based Compressed Video Quality Enhancement. *IEEE Transactions on Broadcasting* (2025).
 - [52] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. IEEE, 586–595.
 - [53] Xindong Zhang, Hui Zeng, and Lei Zhang. 2021. Edge-oriented convolution block for real-time super resolution on mobile devices. In *Proceedings of the 29th ACM international conference on multimedia*. 4034–4043.
 - [54] Minyi Zhao, Yi Xu, and Shuigeng Zhou. 2021. Recursive fusion and deformable spatiotemporal attention for video compression artifact reduction. In *Proceedings of the 29th ACM international conference on multimedia*. 5646–5654.
 - [55] Zhaowen Zheng, Yiqiao Huang, and Dihui Chen. 2024. A lightweight distillation recurrent convolution network on FPGA for real-time video super-resolution. *Multimedia Systems* 30, 6 (2024), 318.