



# Enhancing Video Quality Using a Multi-Domain Spatio-Temporal Deformable Fusion Approach


Garibaldi da Silveira Júnior   [ Federal University of Pelotas | [garibaldi.ds@inf.ufpel.edu.br](mailto:garibaldi.ds@inf.ufpel.edu.br) ]

Gilberto Kreisler  [ Federal University of Pelotas | [gkreisler@inf.ufpel.edu.br](mailto:gkreisler@inf.ufpel.edu.br) ]

Bruno Zatt  [ Federal University of Pelotas | [zatt@inf.ufpel.edu.br](mailto:zatt@inf.ufpel.edu.br) ]

Daniel Palomino  [ Federal University of Pelotas | [dpalomino@inf.ufpel.edu.br](mailto:dpalomino@inf.ufpel.edu.br) ]

Guilherme Corrêa  [ Federal University of Pelotas | [gcorrea@inf.ufpel.edu.br](mailto:gcorrea@inf.ufpel.edu.br) ]

 Video Technology Research Group (ViTech), Universidade Federal de Pelotas (UFPe), Rua Gomes Carneiro - 1, Pelotas-RS, Brazil, 96010-610

**Received:** 31 December 2024 • **Accepted:** 07 June 2025 • **Published:** 06 August 2025

**Abstract** Video compression is essential for efficient data management but often introduces artifacts that degrade visual quality. This work presents the Multi-Domain Spatio-Temporal Deformable Fusion (MD-STDF) architecture, which employs a multi-domain learning approach to enhance the quality of compressed videos across codecs like HEVC, VVC, AV1, and VP9. By training on multiple domains, the model effectively adapts to diverse artifact patterns and compression scenarios. Experimental results show that MD-STDF achieves significant quality improvements, with average  $\Delta$ PSNR gains of 0.764 dB for HEVC, 0.695 dB for VVC, 0.359 dB for VP9, and 0.228 dB for AV1. The model also demonstrated resilience under different compression rates, with BD-Rate values indicating that video quality can be efficiently restored in high compression scenarios for VP9 (-16.50%) and HEVC (-14.59%). Visual analysis shows a reduction in artifacts, resulting in perceptible improvements in subjective quality.

**Keywords:** Video Compression, Video Quality Enhancement, Multi-Domain Learning

## 1 Introduction

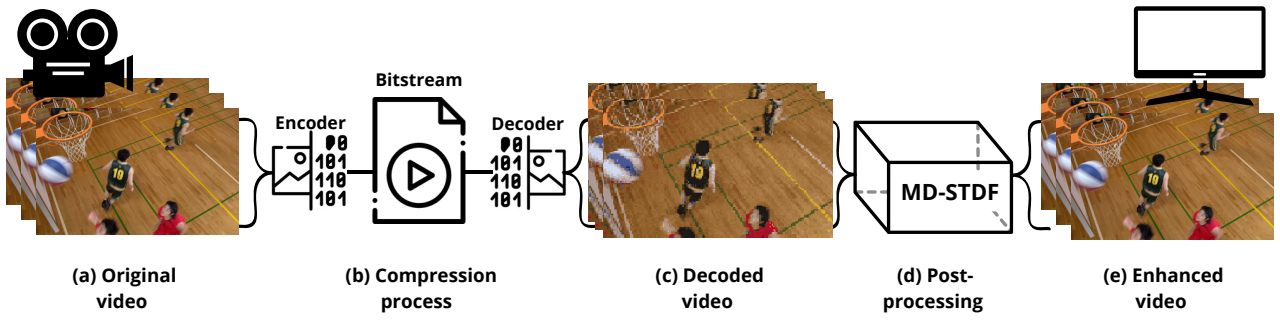
Video compression plays a crucial role in services dealing with the distribution and storage of audiovisual content, becoming essential for the operation of companies like Netflix, TikTok, and YouTube. Due to the high demand for this type of service, digital video represented the highest volume of data transmitted over the Internet in recent years. The Global Internet Phenomena 2023 highlighted that video streaming remains the main driver of global internet traffic growth [Sandvine, 2023]. Consequently, research efforts by both academia and industry are dedicated to improving not only compression efficiency but also reducing undesired visual effects caused by this process. As video content continues to proliferate across various platforms and devices, maintaining high visual quality while efficiently managing data transmission and storage becomes increasingly paramount.

The original video is composed by frames captured by a camera device in pixel format (Figure 1-(a)), typically in the RGB color space, where each pixel is represented by three color channels: red, green, and blue. Each channel is typically encoded using 8 bits. Subsequently, the video frames are divided into macroblocks or coding tree units (CTUs), as defined by the chosen compression standard, such as High Efficiency Video Coding (HEVC). The encoder applies various techniques, including motion estimation, intra-frame prediction, and transform coding, to reduce spatial and temporal redundancies in the video stream [Sullivan *et al.*, 2012]. This process generates a compressed bitstream (Figure 1-(b)), which is a compact, binary representation of the video data

containing all necessary information for decoding. It allows efficient storage or transmission by reducing redundancies while enabling accurate reconstruction during decoding.

The decoding process reconstructs the original video from the compressed bitstream by reversing the encoding steps. In lossy coding methods, compressed videos suffer from visual effects such as blocking, ringing and blurring artifacts [Dong *et al.*, 2015], which compromise the perceived video quality for users (Figure 1-(c)). In Figure 2, the visual effects of these artifacts can be observed. The patches (c), (d) and (e) are separated by the artifact that predominantly affects the selected part of the image. Generally, artifacts can blend with or appear next to others, making it difficult to generate a patch that isolates only one compression artifact. In (c), the division between blocks used in the compression process is perceptible in the middle of the blue part of the ball, evidencing a blocking effect. In (d), we can see the ringing effect, noticeable as wave-like patterns along the edges of the orange rim. In (e), details from the original image, such as the nails on the ground and the separations in the wooden floor, are lost during compression, leading to a blurring effect.

Filtering algorithms like the Deblocking Filter (DF) [Norkin *et al.*, 2012], addressing blocking effects, the Adaptive Loop Filter (ALF) [Tsai *et al.*, 2013], that minimizes the distortion between the original and decoded samples, and the Sample Adaptive Offset (SAO) [Fu *et al.*, 2012], focused on reducing banding effects, are standardized processes in codecs such as High Efficiency Video Coding (HEVC) and Versatile Video Coding (VVC). These algorithms are applied as post-processing methods (Figure 1-(d)), at the end of the



**Figure 1.** Video capture, compression and quality enhancement process: (a) Original video captured from a camera device in pixel format; (b) Compression process, where the original video are encoded to a bitstream format; (c) Decoded video from the bitstream in pixel format; (d) Post-processing method; (e) Enhanced video generated by the post-processing model to a video output.

decoding process and are not part of the decoding loop itself, meaning that the decoded frame is first generated without these enhancements and then filtered for improvement.

Both DF and SAO are heuristic-based methods, devised based upon statistical observations for reducing compression artifacts. These models are applied as filters that traverse all pixels in each frame, aiming to enhance visual quality (Figure 1-(e)). While these heuristic-based approaches have demonstrated effectiveness in mitigating certain artifacts, they have inherent limitations. For example, DF may introduce a blurring effect on the image as it attempts to reduce blockiness, potentially sacrificing fine details and sharpness. Additionally, both DF and SAO may inadvertently amplify the presence of other compression artifacts, such as ringing or mosquito noise, particularly in regions with high contrast or intricate textures [Li *et al.*, 2019]. Despite that, heuristic-based methods remain valuable tools of video compression techniques, especially when used in conjunction with more sophisticated algorithms and Deep Neural Networks (DNN) to achieve comprehensive artifact reduction and enhance overall visual quality.

Currently, a significant amount of studies exploring the Video Quality Enhancement (VQE) problem employ DNN models based on Convolutional Neural Networks (CNN). CNNs have emerged as a powerful tool in image and video processing tasks due to their ability to automatically learn hierarchical features from data. Unlike traditional image processing techniques that operate on individual pixels, CNNs operate by convolving learned filters across the input image, enabling them to capture spatial hierarchies and dependencies. By processing local image patches and learning from their correlations, CNNs can effectively extract meaningful features that represent various visual patterns, such as edges, textures, and object shapes. This enables them to understand the contextual connection between neighboring pixels and learn complex patterns in the data [Li *et al.*, 2021]. Therefore, CNN-based models are well-suited for VQE tasks as they can identify and address overall image quality degradation caused by compression artifacts rather than focusing solely on specific types of artifacts.

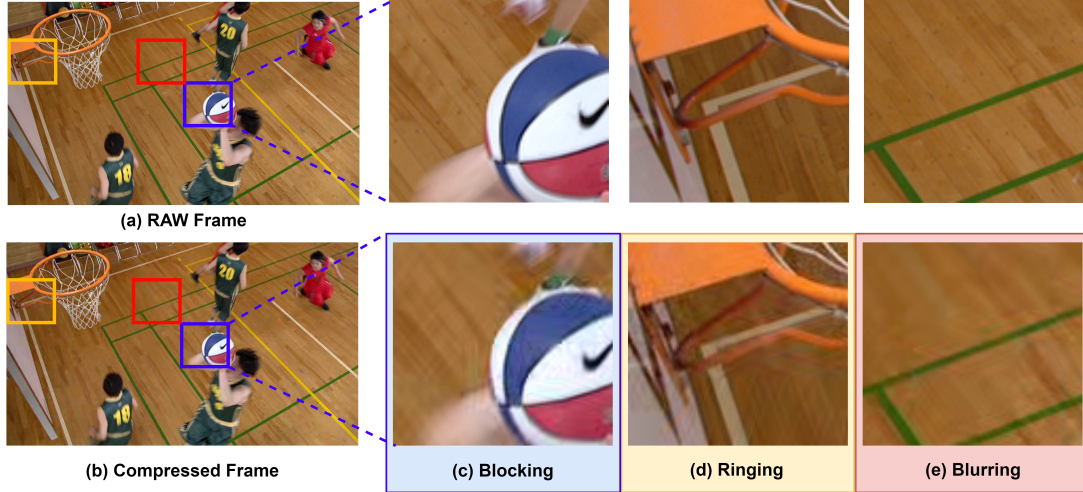
It has been observed that many DNN models for VQE are tested using videos compressed with the same codec and configuration as the training videos. Kreisler *et al.* [2024] show that VQE models tend to produce better results for videos

compressed with the same codec and quantization parameter as those used for training. Oppositely, for videos compressed with different codecs and configurations the VQE models lead to little improvement in quality or even quality degradation. Thus, considering the large number of video codecs and encoding configurations available nowadays, it is desirable that the VQE model at the decoder side is generic enough to be used for enhancing videos compressed in different scenarios.

To address this issue, this work proposes the use of a multi-domain training approach from Chen *et al.* [2018a] that allows identifying the video encoding scenario, generating a model that is adaptive to the video codec and its associated compression artifacts. The proposed Multi-Domain Spatio-Temporal Deformable Fusion (MD-STDF) architecture is used in post-processing and explores multi-domain training for quality improvement of compressed videos, ensuring that a single model is efficient in enhancing the quality of videos originating from any of the domains involved in training. In this work, the videos used in both the training and testing phases have been categorized into domains based on the video codec used for compression: HEVC, VVC, VP9, and AV1.

Experimental results demonstrate that the proposed approach leads to a consistent increase in objective quality for videos compressed with multiple codecs, reaching an average  $\Delta$ PSNR value of up to 0.764 dB for HEVC, 0.695 dB for VVC, 0.359 dB for VP9, and 0.228 dB for AV1. Also, the experiments indicate that video quality can be restored even in high compression rates, with BD-Rate values of -16.50% for VP9 and -14.59% for VVC-encoded videos, thus decreasing the bitrate required for video transmission and storage without affecting the user quality of experience. To the best of the authors' knowledge, this is the first DNN architecture to employ multi-domain training for VQE and that has been trained and tested for multiple video codecs and formats.

This paper is organized as follows. Section 2 presents previous VQE works. Section 3 presents the proposed Multi-Domain Spatio-Temporal Deformable Fusion (MD-STDF) architecture. Section 4 presents and discusses the obtained objective results and Section 5 presents a discussion on subjective visual quality perception. Section 6 presents a computational cost analysis on the proposed model and Section 7 concludes this work.



**Figure 2.** Compression artifacts: (a) Original frame (RAW); (b) Compressed frame; (c) Blocking artifact; (d) Ringing artifact; (e) Blurring artifact.

## 2 Related Work

In recent years, different architectures have been proposed to investigate the VQE problem. In this section, we present the main studies that have contributed to the evolution of learning-based models for VQE. Also, related works focusing on multi-domain learning for video-related problems are discussed.

### 2.1 Video Quality Enhancement

Solutions for VQE emerged with the application of frame-by-frame image processing algorithms in videos. These methods originated from linear algorithms based on heuristics, processing all pixels in the frame, disregarding spatial differences within the same frame, and applying the same equation to the entire image. While this approach effectively addressed degradation issues concentrated in specific parts of the image, it often detrimentally affected other areas that did not share the same problem.

These algorithms have gradually given way to nonlinear machine learning models, which aim to comprehend a zone of pixels and detect their characteristics, thereby determining which heuristic yields superior results in a given scenario. Consequently, the side effects caused by these processes tend to diminish. Moreover, the advent of DNN has led to the development of more robust architectures for nonlinear learning, enabling the detection of increasingly intricate characteristics as network depth increases. This evolution has significantly contributed to the development of more effective methods for VQE.

One of the first solutions for improving video quality based on deep learning is [Dong *et al.*, 2015], which proposes an Artifact Reduction CNN (ARCNN) that processes each frame individually, exploring only the spatial information within the image. Based in the work of Dong *et al.* [2015], other applications for the model were explored, such as an in-loop filter that replaces DF and SAO filters [Dai *et al.*, 2017b] or as a post-processing filter [Kuanar *et al.*, 2018], which performs the filtering after the frames are fully decoded.

Some studies emerged with the proposal of exploring the existing temporal correlation between frames. Initially, mod-

els based on multiple frames [Yang *et al.*, 2018b; Guan *et al.*, 2019; Caballero *et al.*, 2017; Deng *et al.*, 2020] proved effective for the VQE problem. These models define a temporal sliding window that processes a fixed number of frames to improve the central one. This way, the Group of Pictures (GOP) structure present in most of the video coding standards can be explored, allowing information from high-quality frames to be used to improve low-quality frames [Tong *et al.*, 2019].

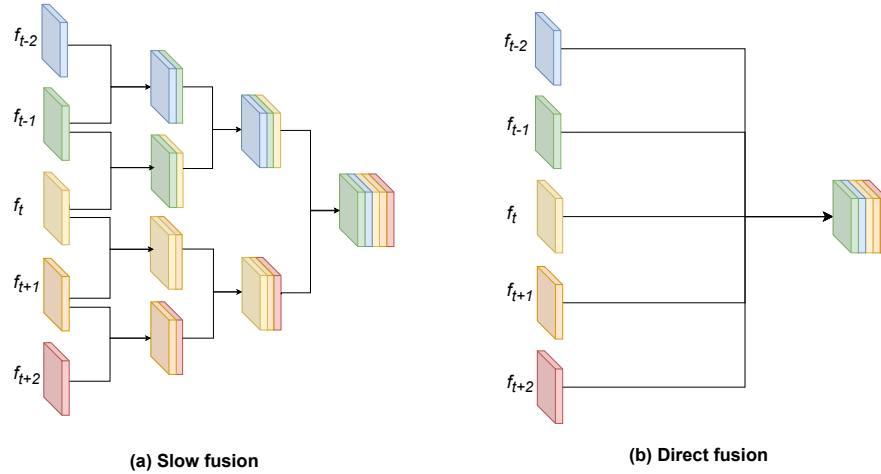
Models based on multiple frames aim to synchronize past and future frames with the currently processed frame. This alignment process is typically achieved through motion compensation techniques, such as optical flow estimation [Xue *et al.*, 2019; Tong *et al.*, 2019]. Optical flow determines motion vectors between consecutive frames by calculating pixel displacements, enabling the model to understand temporal relationships and align frames effectively. By comparing corresponding points in neighboring frames, it estimates pixel movement along the horizontal and vertical axes, facilitating motion compensation. This alignment preserves temporal continuity and smooth transitions in video quality enhancement, reducing artifacts like ghosting or blurring.

Following the alignment process, the fusion of processed frames occurs, with the goal of incorporating the best quality characteristics from each frame. Two common fusion approaches include direct fusion and slow fusion. As can be seen in figure 3, slow fusion (a) involves iteratively fusing pairs of frames until only one frame remains, gradually synthesizing the final enhanced frame. On the other hand, in direct fusion (b), all frames are fused simultaneously, leveraging the information from each frame to enhance the overall quality [Meng *et al.*, 2019].

This alignment and fusion strategy allows models to exploit temporal information across multiple frames, leading to more comprehensive and effective video quality enhancement. By aligning frames and fusing their features, these models can better capture and preserve the temporal coherence and visual details present in the video sequence.

An alternative to the optical flow-based fusion is the use of deformable convolutions [Wang *et al.*, 2019; Deng *et al.*, 2020]. This mechanism replaces the standard convolution





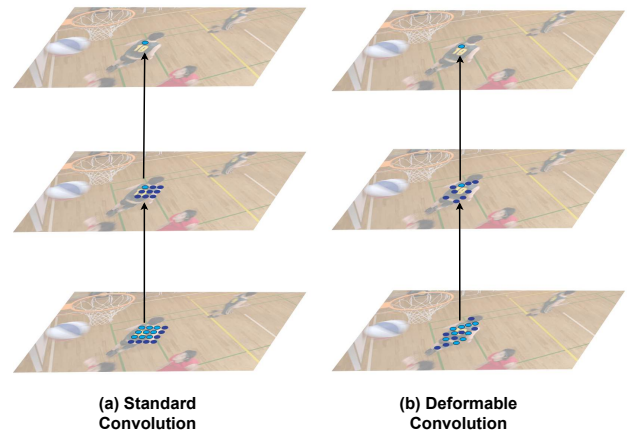
**Figure 3.** Fusion approaches in a range of two neighbor frames and a central one: (a) Slow fusion, where the neighboring frames ( $f_{t-2}$ ,  $f_{t-1}$ ,  $f_{t+1}$ , and  $f_{t+2}$ ) are processed sequentially, gradually integrating their features with the central frame ( $f_t$ ) to refine temporal dependencies; (b) Direct Fusion, where the frames ( $f_{t-2}$  through  $f_{t+2}$ ) are fused in a single operation.

used by CNN layers with a deformable convolution. As shown in Figure 4, instead of using a fixed, regular sampling grid for the convolutional kernel, deformable convolutions dynamically adjust the sampling locations by introducing learnable offsets. These offsets are predicted from the input feature map using additional convolutional layers, allowing the kernel to adapt its receptive field to the spatial characteristics of the data.

In standard convolutions (Fig. 4(a)), the sampling grid is fixed and predetermined, with kernel elements centered on the target pixel and evenly spaced around it. This rigid structure limits the ability of the model to effectively capture deformations, motion, or other non-uniform patterns in the input. Conversely, deformable convolutions (Fig. 4(b)) overcome this limitation by learning offsets that enable the kernel to sample from irregular or distant pixel locations. This flexibility allows the network to focus on semantically relevant regions, even when they do not align perfectly with the grid, thus improving the model's ability to handle complex spatial variations.

From a computational perspective, traditional convolutions are simpler and faster, as they only involve regular grid sampling and straightforward matrix multiplications. Deformable convolutions, however, require additional computation to predict the offsets and to sample pixel values from non-integer coordinates using interpolation (e.g., bilinear interpolation) [Dai et al., 2017a]. While this increases computational cost, the benefits in terms of feature alignment and robustness to spatial transformations make deformable convolutions particularly effective for tasks such as video enhancement and object detection, where precise handling of motion and spatial variability is critical. By eliminating the need for explicit motion estimation, deformable convolutions offer a compelling alternative to optical flow-based techniques, which rely on computationally expensive and often error-prone processes to align features across frames [Mac et al., 2019].

Another strategy employed to harness temporal information in video processing involves the utilization of recurrent models, particularly Recurrent Neural Networks (RNNs).



**Figure 4.** Types of convolution: (a) Fixed matrix of pixels 3x3 used in standard convolution, where the kernel samples pixel values in a fixed and uniform grid pattern across the input feature map; (b) Matrix with displaced points used in deformable convolution, which adaptively shifts sampling locations to better align with the geometric structure of the input.

RNNs are designed to handle sequential data by processing input sequences one step at a time, maintaining a hidden state that captures information from previous steps [Liang and Hu, 2015]. In the context of video quality enhancement, RNNs operate by progressively analyzing each frame of the video sequence. As each frame is processed, its characteristics are extracted and incorporated into the hidden state of the network. This hidden state serves as a condensed representation of the temporal information extracted from the video sequence so far, enabling the network to understand the temporal dependencies and patterns present in the data. Subsequently, this aggregated temporal information can be utilized to enhance the quality of future frames through filtering or other enhancement techniques. Studies explore recurrent networks in different ways, either in a unidirectional manner [Nah et al., 2019], where the characteristics are propagated from past frames to the currently processed frame, or using a bidirectional improvement [Zhu et al., 2022], where both features from past and future frames are used to improve

the quality of the current frame.

## 2.2 Multi-Domain Learning

The advancement of machine learning algorithms has been made possible due to the abundance of available data. However, the current training paradigms are limited in the variety of data they can handle. Most methods operate on data from specific domains, causing the model to learn the inherent bias of the dataset. As a result, the models perform well on tasks specific to the domains they were trained on, sometimes limiting their ability to generalize when processing data from new and previously unseen domains. These challenges become more pronounced when dealing with data from highly variable domains, especially when there is a need to develop a single model capable of handling multiple distinct datasets. Training a single model to encompass this diversity of domains prevents the capture of specific nuances from each one. The common approach to address this issue involves recreating a model for each domain and applying each model to the corresponding data. However, this methodology proves to be highly inefficient.

Videos exhibit characteristics with a high variability, making it challenging for a model to generate a unique representation that can capture them all. Therefore, the Multi-Domain Network (MDNet) of Nam and Han [2016] emerged with the proposal of separating videos into annotated domains, so that each domain follows a distinct path in the final layers of the network. In this way, common features among all domains are extracted in the initial layers of the network, while features specific to the domain are extracted in the final layers. Following the same proposal as the previous study, the Branch-Activated Multi-Domain Convolutional Neural Network for Visual Tracking (BAMCNN) was developed by Chen *et al.* [2018a]. It leverages the concept of multi-domain training to create an architecture for visual tracking, separating videos into different domains based on their similarities. This involves creating branches in the network for each domain, thus detecting their specific characteristics. During testing, the branch with the highest level of similarity to the processed video sequence is identified and activated.

Recently, other studies have employed multi-domain learning for different video-related problems. In the work of Peng *et al.* [2022], the authors addressed the quality enhancement of multiview video coding. The work of Agarwal *et al.* [2021] explores the detection of deepfake videos by combining features from two distinct domains, spatial and frequency, in a discriminative learning model.

From the related works discussed, it is possible to observe that most existing VQE approaches are designed and optimized for a specific video coding standard, such as HEVC. In this context, each coding standard can be interpreted as a distinct domain, given that different encoders introduce different types and distributions of artifacts. As such, solutions trained within a single domain tend to specialize in handling the artifacts characteristic of that particular standard. The model proposed in this work differentiates itself by being trained on videos compressed with four major standards – HEVC, VVC, VP9, and AV1 – treating each as a separate domain.

## 3 Multi-Domain STDF

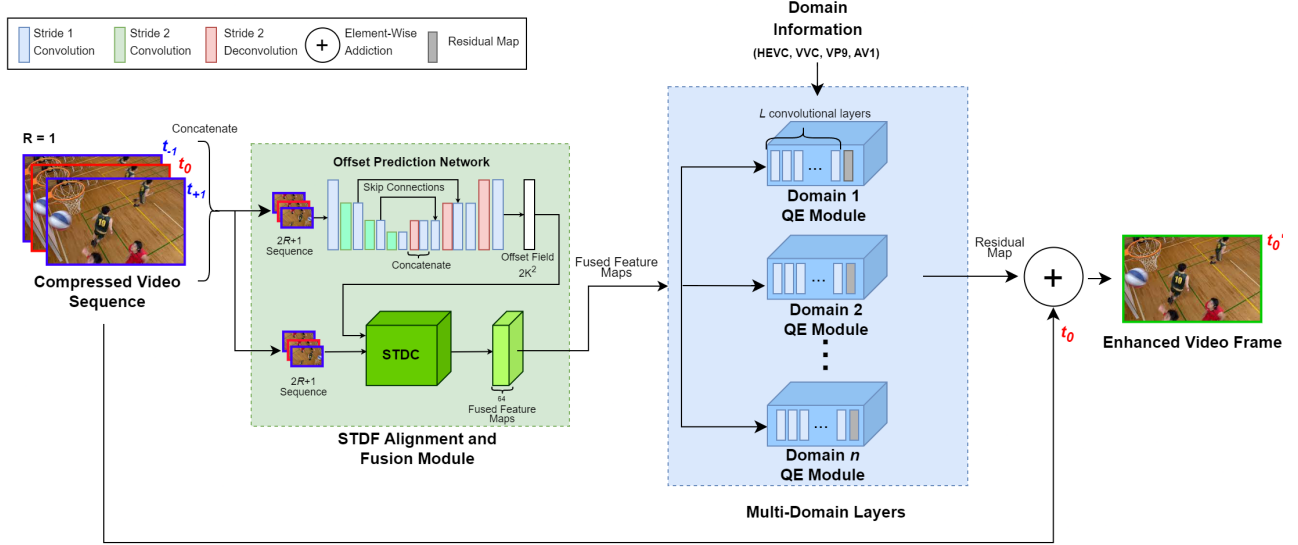
The proposed Multi-Domain Spatio-Temporal Deformable Fusion (MD-STDF) architecture is based on the STDF architecture of Deng *et al.* [2020], which employs the approach of using multiple frames to enhance a central frame. Additionally, the architecture applies the multi-domain training strategy, with training data originated from different domains. The domain information is also incorporated in the dataset to allow the model to learn the domain-specific characteristics along the training.

The STDF architecture is divided into two modules, with the first one focusing on frame alignment and fusion, as well as shallow feature extraction. The second module is dedicated to quality enhancement. The model receives as input the central frame to be improved ( $t_0$ ), concatenated with temporally neighboring frames ( $t_1$  for the future frames and  $t_{-1}$  for past frames). The number of neighboring frames to be concatenated with the central frame is determined by the Radius ( $R$ ) parameter, which represents the number of neighboring frames. Thus, with  $R$  defined as 1, the total number of frames to be concatenated and introduced into the model is 3.

As depicted in Figure 5, in the development of MD-STDF, the alignment and fusion module is dedicated to obtaining general features — i.e., those that are common to all videos regardless of the domain. The network starts with a shallow U-Net Model [Ronneberger *et al.*, 2015] that extracts the features for offset prediction. This model is based on Stride 1 convolutions, which maintain the sample dimensionality, Stride 2 convolutions, which reduce sample dimensionality, and Stride 2 deconvolutions, which perform an upsampling. This part of the network captures the temporal characteristics of the sequence and generates an offset field mask. The kernel size of the offset mask is  $2K^2$ . This mask is used with the input  $2R+1$  sequence in the Spatio-Temporal Deformable Convolution (STDC) network, generating a Fused Feature Map with 64 channels.

Instead of calculating the difference between two frames, which is the method used in optical flow, in studies like [Guan *et al.*, 2019], the STDC module uses a modulated deformable convolution layer [Zhu *et al.*, 2019]. This layer performs the motion compensation of the entire input sequence at once. As an alternative for using a fixed grid of positions to apply the convolutional kernels (as in conventional convolutions), deformable convolution introduces additional offsets that are learned by the network during training. These offsets allow the convolutional kernel to “deform” to better align with the input patterns, resulting in improved capture of spatial and temporal variations [Dai *et al.*, 2017a].

The multi-domain training is guided by a label that links each video to a specific domain (codec), and this label is used in the transition between the alignment and fusion module and the quality enhancement (QE) module. Dedicated branches are created for each domain to ensure that the QE module is updated with parameters specific to the corresponding codec. The activated branch delivers the Fused Feature Map to the appropriate QE module based on the video label. Each QE module consists of  $L$  convolutional layers with stride 1, preserving the spatial resolution of the input



**Figure 5.** The MD-STDF architecture consists of the STDF Alignment and Fusion Module, that is shared by all domains and  $n$  branches of QE multi-domain. The red-border frame represents the central one, under processing, whereas blue-border neighboring frames are aligned and used in the enhancement of the central frame.

and focusing on the extraction of codec-specific features. the end of the process, the QE module generates a residual map, which is added to the central frame ( $t_0$ ) to produce the enhanced frame ( $t_0'$ ). This residual map is a single-channel (luminance) feature map of dimensions  $H \times W$ , representing pixel-wise corrections predicted by the model. It is applied via an element-wise addition to the central frame, introducing minimal computational and memory overhead. This process is repeated for each frame in the compressed video, resulting in a fully enhanced video sequence.

### 3.1 Data Preparation

A survey of the main datasets used for VQE was first conducted to decide the dataset used for training. This survey did not include image datasets, only videos. Additionally, datasets with a specific purpose, such as screen content videos or sports videos only, were not included, but rather those that cover a wide category range. Among the possible choices for datasets are the MFQE dataset used in [Guan et al., 2019], the Large-scale Diverse Video (LDV) dataset used in the challenge proposed at the New Trends in Image Restoration and Enhancement workshop and challenges on image and video processing (NTIRE) in 2021 [Yang and Timofte, 2021], and the Vimeo-90K dataset [Xue et al., 2019].

MFQE dataset [Yang et al., 2018a] was selected due to its widespread use in the VQE field, serving as a common benchmark in numerous previous studies [Guan et al., 2019; Deng et al., 2020; Yang et al., 2018a; Chen and Ye, 2021; Zhao et al., 2021a]. It contains 126 uncompressed videos at different resolutions, ranging from  $352 \times 240$  to  $1920 \times 1080$ . Among these, 108 videos were designated for training and 18 for testing. This split was originally defined by the creators of the dataset and was preserved in this work to ensure greater reproducibility of the results. Furthermore, the same test set was used during the validation phase, as commonly adopted in related works, allowing consistent performance tracking throughout the training process.

The video sequences were organized according to the standards used for compression, in order to split the dataset for the multi-domain training. Four versions of the training dataset were generated, each corresponding to the addressed domains. Thus, the 108 videos were encoded and decoded using the reference software for the four standards/formats, resulting in a total of 432 videos. The four distinct domains defined correspond to the High Efficiency Video Coding (HEVC) standard [Sullivan et al., 2012], the Versatile Video Coding (VVC) standard [Bross et al., 2021], the AOMedia Video 1 (AV1) format [Chen et al., 2018b], and the VP9 [Mukherjee et al., 2013] format.

For HEVC encoding, the reference software HEVC Model (HM) version 16.5 was used, and for VVC, the VVC Test Model (VTM) version 13.0 was employed, both configured with *Low Delay P* temporal setting. For AV1 encoding, the reference software *libaom*, hashcode 3.3 was used, whereas for VP9 the *libvpx*, hashcode 1.12.0, was used. For HEVC and VVC, the quantization parameter (QP) was set to 37, while for VP9 and AV1, constant quality (CQ) parameter was set to 55. The QP/CQ controls the level of quantization applied to the transformed residual blocks. The use of high QP/CQ values causes information to be discarded during the quantization process, leading to a loss of fine details and, consequently, lower image quality. On the other hand, lower QP/CQ values result in less quantization and superior visual quality. According to the Bender et al. [2019] analysis, there is no correlation between the QP and CP values that indicates the same level of quality loss between both encoders. Thus, CQ and QP values were chosen as those that lead to the highest level of degradation according to the recommended test conditions of HEVC/VVC and VP9/AV1.

### 3.2 Training Process

For training, we randomly crop  $128 \times 128$  clips from raw and the corresponding compressed videos as training samples. Data augmentation techniques such as rotation and flipping

are employed to better exploit the available data. All models are trained using the Adam optimizer, with the learning rate initially set to  $10^{-4}$  and retained throughout training. The Charbonier loss function [Kingma and Ba, 2014] is used in our experiments, as it focuses on minimizing pixel-level reconstruction errors. The training was conducted on a machine equipped with an AMD Ryzen 7 5700X processor, 32 GB RAM, and an Nvidia GeForce RTX 3070 GPU with 8 GB VRAM. To complete 10 epochs using a single GPU, we set the batch size to 32 and performed 1,200,000 iterations over the dataset. For evaluation, we apply quality enhancement only on the Y-channel (i.e., luminance component) in the YUV/YCbCr color space.

The algorithm employed was Stochastic Gradient Descent (SGD) [Nam and Han, 2016], where each training iteration was executed under a specific domain. In other words, the batch of videos used belongs to a single domain, activating only one branch of the network, as shown in Figure 5. After a certain number of iterations, the model is updated based solely on the processed batch. Subsequently, a new batch from another domain is processed, causing the shared layers of the network to be updated while keeping the previously updated branch unchanged. This process is repeated until the predefined number of iterations is reached. Following this approach, the generic features common to all processed videos are obtained in the shared layers of the network, while for each specific branch of each domain, modeling is done to acquire domain-specific characteristics.

## 4 Experimental Results

### 4.1 VQE Results

The obtained results offer a comprehensive view of the conducted study and aid in assessing the effectiveness of the adopted multi-domain training methodology. The results are presented in terms of  $\Delta$ PSNR, which measures the objective difference between the enhanced and the low-quality decoded video. Positive numbers indicate an increase in objective quality, whereas negative numbers indicate a quality decrease.

For comparison purposes, three single-codec STDF models were also trained following the methodology presented in Deng *et al.* [2020]: the first with a dataset containing only videos compressed with the HEVC codec; the second with videos compressed with the VVC codec; and the third with videos compressed with the AV1 codec. Additionally, the multi-codec, single-domain approach proposed in Kreisler *et al.* [2024] is also presented. The same encoder configurations and training setup mentioned in the previous section were used.

Table 1 shows the results summary for comparison purposes. The first three rows (HEVC QP37, VVC QP 37 and AV1 CQ 55) present the average VQE results obtained from training models using the single-codec approach. In the fourth row, the results obtained from training using the multi-codec approach from Kreisler *et al.* [2024] are shown. Finally, the last two rows presents the results obtained with the models trained using the proposed multi-

domain method. The first model, named “Multi-Domain low QP/CQ” is trained with videos compressed by HEVC and VVC with QP 32, and by VP9 and AV1 with CQ 43. The second model, named “Multi-Domain high QP/CQ”, is trained with videos compressed with the same codecs, but using QP 43 and CQ 55, respectively.

As observed, the model trained with videos compressed with HEVC achieves the best results (0.755 dB) when tested with videos encoded with the same standard. This highlights the strong compatibility of single-codec models when applied to their corresponding compression standard. However, for videos encoded with the AV1 standard, the model yields a negative VQE result (-0.506 dB), indicating a clear limitation in generalizing to different compression schemes. A similar trend is observed for the model trained with videos compressed with VVC, which performs poorly for AV1-compressed videos. These results underline that single-codec models, while effective within their specific domain, face challenges in handling cross-codec scenarios, likely due to the inherent differences in compression strategies and distortions introduced by each codec.

The multi-codec model proposed by Kreisler *et al.* [2024] presents more constant results (varying between 0.210 dB and 0.375 dB), demonstrating better generalization across codecs compared to single-codec models. However, it does not perform as well as the single-codec models in their optimized domains, which suggests a trade-off between generalization and peak performance. Finally, the proposed multi-domain model is the one that presents the best results in all cases (varying between 0.228 dB for AV1 and 0.764 dB for HEVC in the high QP/CQ model, and 0.354 dB for AV1 and 0.697 dB for HEVC in the low QP/CQ). This demonstrates its ability to adapt effectively to various codecs while maintaining high performance. Notably, the Multi-Domain model trained with high QP/CQ configuration performs better for videos with high QP/CQ settings, indicating that this configuration is more suited to handling higher levels of compression. On the other hand, the low QP/CQ model also achieves strong results, with less than  $\pm 0.1$  dB of difference across all scenarios, having a superior performance with AV1-encoded videos, where it achieves a PSNR value of 0.354 dB compared to 0.228 dB for the high QP/CQ model.

Table 2 presents more detailed results of the obtained objective quality variation with the “Multi-Domain high QP/CQ”. The results are presented separately for each video sequence for QP 37 and CQ 55, i.e. the same used for training the model. For the remaining three pairs of QP/CQ, average values are shown at the bottom of the table. Table 3 presents similar data, but using the “Multi-Domain low QP/CQ” model. The videos are grouped according to their Class [Boyce *et al.*, 2018]: Class A (2560x1600), Class B (1920x1080), Class C (832x480), Class D (416x240), Class E (1280x720).

To validate the hypothesis that the proposed approach enhances the quality of videos compressed under different codec configurations and QP/CQ values, we selected compression parameters that closely approach the maximum values supported by each codec. For HEVC and VVC, in addition to test videos compressed with QP values of 32 and 37, videos were also compressed using higher QP values of 42

**Table 1.** Comparison between single-codec, multi-codec and multi-domain approaches.

STDF Model	$\Delta$ PSNR (dB)			
	HEVC	VVC	VP9	AV1
	QP 37	QP 37	CQ 55	CQ 55
HEVC QP 37 [Deng et al., 2020]	0.755	0.250	0.357	-0.506
VVC QP 37	0.529	0.371	0.385	-0.016
AV1 CQ 55	0.285	0.144	0.389	0.286
Multi-Codec [Kreisler et al., 2024]	0.335	0.210	0.375	0.229
<b>Multi-Domain low QP/CQ</b>	<b>0.697</b>	<b>0.413</b>	<b>0.675</b>	<b>0.354</b>
<b>Multi-Domain high QP/CQ</b>	<b>0.764</b>	<b>0.448</b>	<b>0.736</b>	<b>0.228</b>

and 47. For VP9 and AV1, beyond the CQ 43 and 55 configurations, the videos were also encoded with CQ values of 60 and 63. This strategy allows evaluating the model's performance in extreme compression scenarios, where the visual quality of the video is significantly degraded due to low target bitrates for transmission and storage. By analyzing these scenarios, we can better understand the robustness of the model in handling severe compression artifacts and ensuring meaningful quality enhancement, even when the bitstream is drastically constrained.

The objective VQE results in both tables are also presented in terms of Structural Similarity Index Measure ( $\Delta$ SSIM) and Perceptual Image Patch Similarity ( $\Delta$ LPIPS) metrics to complement the  $\Delta$ PSNR values. SSIM quantifies structural similarity by focusing on luminance, contrast, and structure, providing a value between 0 and 1, where higher values indicate better quality [Wang et al., 2004]. LPIPS, on the other hand, employs deep learning models to estimate perceptual similarity, correlating more closely with human visual perception [Zhang et al., 2018]. While PSNR emphasizes numerical differences, SSIM and LPIPS account for structural and perceptual aspects, offering a more comprehensive quality assessment.

As can be observed in the Table 2, most of the results are positive, with only one specific case showing a negative  $\Delta$ PSNR value (-0.024 dB for AV1, *BQTerrace* sequence). However, in terms of  $\Delta$ SSIM, the result is positive for this case. On average, almost all results are positive independently on the metric used, as shown in the last four rows of the table. The best result in terms of  $\Delta$ PSNR is 1.234 dB for VP9 in the *BQSquare* sequence. In terms of  $\Delta$ SSIM, the best result is 0.020 for HEVC in the *People on Street* sequence. Considering average  $\Delta$ PSNR results, the worst improvement is -0.256 dB for videos encoded with AV1 CQ 43 (the only negative average result), and the best improvement is 0.764 dB for videos encoded with HEVC QP 37. Considering average  $\Delta$ SSIM, the worst improvement is 0 dB (insignificant change) for videos encoded with AV1 CQ 43, and the best one is 0.019 for videos encoded with HEVC QP 42. In terms of  $\Delta$ LPIPS, we can see that the worst average enhancement value is 0.001, in videos compressed with VVC with QP 42 and HEVC with QP 47. The best average LPIPS enhancement value is for VP9 with CQ 43 (0.040 dB).

Some specific cases, in addition to the best case, showed results in terms of  $\Delta$ PSNR above 1 dB, such as the *People on Street* sequence encoded with HEVC QP 37, which achieved

a result of 1.126 dB; the *BQSquare* sequence encoded with HEVC QP 37 achieved a result of 1.020 dB; and the *FourPeople* sequence encoded with VP9 CQ 55 achieved a result of 1.046 dB.

These results demonstrate the model's effectiveness in enhancing video quality, with consistently positive average improvements in  $\Delta$ PSNR,  $\Delta$ SSIM, and  $\Delta$ LPIPS metrics across most cases. Notable performance was observed with HEVC and VP9, particularly at lower QP or CQ settings, while challenges emerged with AV1, including occasional negative  $\Delta$ PSNR values.

Table 3 present the results of the "Multi-Domain low QP/CQ" model over test videos. As can be seen, there are no average negative values in any of the metrics used to evaluate the test dataset. The highest enhancements occur at the lowest QP/CQ configurations, with the best average improvement being 0.787 dB for HEVC QP 32. The individual video with the best  $\Delta$ PSNR value is *VQSquare*, achieving 1.433 dB. The worst average  $\Delta$ PSNR result is observed for VVC at the highest QP configuration (QP 47, 0.112 dB), while the worst individual case is *BQTerrace*, with 0.102 dB in AV1.

In terms of  $\Delta$ SSIM, the lowest average value is 0.004 dB for AV1 at CQ 55, while the highest average value is 0.014 dB for HEVC at QP 37 and 42. The best individual result is for *BlowingBubbles*, achieving 0.018 dB, whereas the worst result is 0.002 dB, occurring in five videos compressed with AV1. Regarding the LPIPS metric, the worst average case is observed in VVC at QP 37 and 42 (0.003 dB), and the best average result is 0.023 dB for VP9 at CQ 55. For individual videos, the worst LPIPS value is -0.011 dB for *BQSquare* in AV1, while the best is 0.048 dB for *Cactus* in VP9. These results indicate that the model performs better at lower QP/CQ configurations, particularly for HEVC and VP9, which show consistent positive enhancements across all metrics. AV1 demonstrates some variability with both the lowest and highest individual values, while VVC generally shows modest improvements.

When comparing the results from the high QP/CQ and low QP/CQ configurations, it is evident that the Multi-Domain model achieves better overall performance at lower QP/CQ settings. The low QP/CQ results show consistently higher enhancements, particularly in  $\Delta$ PSNR, where HEVC and VP9 exhibit significant improvements, such as the 0.787 dB average for HEVC at QP 32 and the 1.433 dB individual enhancement for *VQSquare*. In contrast, the high QP/CQ results are more variable, with notable challenges for AV1, including



**Table 2.** VQE results for the MD-STDF high QP/CQ model ( $\Delta$ PSNR,  $\Delta$ SSIM and  $\Delta$ LPIPS).

Configuration	Video Resolution Class	Video	VQE Results ( $\Delta$ PSNR, $\Delta$ SSIM and $\Delta$ LPIPS)											
			HEVC			VVC			VP9			AV1		
			PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS
QP 37 / CQ 55	Class A	Traffic	0.663	0.011	0.024	0.420	0.006	0.005	0.727	0.009	0.019	0.120	0.003	0.017
		PeopleOnStreet	1.126	0.020	0.006	0.581	0.009	0.003	0.911	0.014	0.023	0.200	0.004	0.024
	Class B	Kimono	0.828	0.016	0.042	0.548	0.008	0.002	0.462	0.007	0.029	0.184	0.003	0.017
		Parkscene	0.554	0.013	0.029	0.426	0.010	0.001	0.487	0.008	0.023	0.152	0.004	0.017
		Cactus	0.696	0.013	0.026	0.404	0.007	0.008	0.641	0.010	0.032	0.129	0.003	0.021
		BQTerrace	0.541	0.009	0.029	0.217	0.004	0.007	0.403	0.006	0.025	-0.024	0.002	0.017
		BasketballDrive	0.689	0.012	0.033	0.292	0.005	0.008	0.552	0.008	0.037	0.156	0.003	0.020
		RaceHorses	0.454	0.011	0.026	0.220	0.006	0.009	0.473	0.011	0.027	0.153	0.003	0.025
	Class C	BQMall	0.843	0.017	0.014	0.529	0.009	0.000	0.828	0.012	0.017	0.231	0.004	0.015
		PartyScene	0.637	0.019	0.025	0.381	0.011	0.005	0.731	0.014	0.026	0.197	0.005	0.020
		BasketballDrill	0.721	0.014	0.013	0.290	0.005	0.005	0.732	0.014	0.024	0.197	0.004	0.021
		RaceHorses	0.694	0.018	0.005	0.436	0.011	0.000	0.661	0.015	0.008	0.344	0.008	0.018
	Class D	BQSquare	1.023	0.015	-0.001	0.667	0.009	-0.002	1.234	0.011	0.005	0.487	0.004	0.006
		BlowingBubbles	0.632	0.019	0.018	0.453	0.015	0.001	0.702	0.015	0.020	0.328	0.009	0.016
		BasketballPass	0.968	0.019	0.006	0.716	0.015	-0.007	0.859	0.015	0.006	0.496	0.008	0.013
		FourPeople	0.907	0.011	0.010	0.548	0.006	0.006	1.046	0.008	0.014	0.325	0.003	0.014
	Class E	Johnny	0.797	0.007	0.018	0.413	0.003	0.011	0.844	0.006	0.020	0.164	0.002	0.014
		KristenAndSara	0.970	0.009	0.016	0.515	0.004	0.007	0.954	0.006	0.020	0.270	0.002	0.014
		Average	0.764	0.014	0.019	0.448	0.008	0.004	0.736	0.011	0.021	0.228	0.004	0.017
QP 32 / CQ 43	Average		0.399	0.004	0.037	0.253	0.004	0.011	0.401	0.005	0.040	-0.256	0.000	0.026
QP 42 / CQ 60	Average		0.662	0.019	0.006	0.401	0.009	0.001	0.760	0.014	0.012	0.478	0.010	0.005
QP 47 / CQ 63	Average		0.416	0.018	0.001	0.242	0.008	0.002	0.665	0.016	0.003	0.376	0.011	0.002

occasional negative  $\Delta$ PSNR values, although the best enhancement of 1.234 dB for *BQSquare* in VP9 is competitive with the best low-QP results. Moreover, while  $\Delta$ SSIM and  $\Delta$ LPIPS improvements remain consistently positive in both cases, the magnitude of enhancement is generally greater in the low-QP scenario. However, despite these differences, the model demonstrates its capability to improve video quality even at higher QP/CQ settings, where compression artifacts are more pronounced.

## 4.2 BD-Rate and BD-PSNR Results

The efficiency of the proposed multi-domain model across different codecs and QP/CQ configurations can be also analyzed in terms of coding efficiency using BD-Rate and PD-PSNR, metrics introduced by Bjontegaard [2001] to assess the efficiency of video compression algorithms. BD-Rate quantifies the percentage change in bitrate required to maintain a given level of video quality. Alternatively, BD-PSNR quantified the quality change (in dB) when the bitrate is maintained. In other words, BD-Rate compares the bitrate at which two different methods can achieve the same quality, and BD-PSNR compares the quality loss when two different methods employ make use of the same bitrate. Lower BD-Rate values and higher BD-PSNR values indicate better compression efficiency. This metrics allow for a more comprehensive comparison of compression performance across different settings.

The charts in Figure 6 illustrate this comparison in terms of BD-PSNR, but also present BD-Rate values. Data points

in the charts indicate the QP/CQ configurations: QP values of 32, 37, 42, and 47 for HEVC and VVC; CQ values of 43, 55, 60, and 63 for AV1 and VP9. Each chart presents results comparing the decoded video (no VQE used), the video after the use of the baseline STDF model for VQE, and the video after the use of the proposed MD-STDF model for VQE, as red, green and blue curves, respectively.

Notice that in most of the cases the blue line is above the red and green curves, indicating that MD-STDF always achieves superior image quality (PSNR, in dB) considering the same bitrate (in kbps). Each chart also presents the calculated BD-Rate value for STDF and MD-STDF, indicating the required bitrate variation to achieve the same objective quality.

The lowest BD-Rate values (i.e., the best results) are noticed for VP9 and HEVC. When using the “High QP/CQ MD-STDF” model, VP9 requires, on average, -16.50% fewer bits to represent the video sequences without quality loss in comparison to the case in which no VQE model is used. Considering the baseline MD-STDF, a bitrate reduction of only -8.66% is achieved.

The BD-Rate values obtained for MD-STDF-enhanced videos are generally higher compared to STDF-enhanced videos in most cases. The only exception occurs in the HEVC codec with low QP/CQ, in which MD-STDF shows BD-Rate of -12.47% and the original STDF achieves BD-Rate of -14.28%. This can be attributed to the fact that low QP/CQ (QP 32) are used for training MD-STDF in this case, while the original STDF was trained with videos encoded

**Table 3.** VQE results for the MD-STDF low QP/CQ model ( $\Delta$ PSNR,  $\Delta$ SSIM and  $\Delta$ LPIPS).

Configuration	Video Resolution Class	Video	VQE Results ( $\Delta$ PSNR, $\Delta$ SSIM and $\Delta$ LPIPS)											
			HEVC			VVC			VP9			AV1		
			PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS
QP 32 / CQ 43	Class A	<i>Traffic</i>	0.730	0.008	0.024	0.480	0.005	0.005	0.697	0.006	0.022	0.290	0.002	0.008
		<i>PeopleOnStreet</i>	0.920	0.011	0.021	0.465	0.006	0.008	0.741	0.008	0.027	0.298	0.003	0.014
	Class B	<i>Kimono</i>	0.776	0.011	0.043	0.499	0.006	0.004	0.465	0.006	0.037	0.237	0.003	0.013
		<i>Parkscene</i>	0.631	0.011	0.022	0.509	0.009	0.000	0.529	0.007	0.022	0.255	0.003	0.008
		<i>Cactus</i>	0.642	0.009	0.029	0.391	0.006	0.008	0.553	0.007	0.048	0.168	0.002	0.016
		<i>BQTerrace</i>	0.497	0.007	0.027	0.244	0.004	0.009	0.334	0.005	0.029	0.102	0.002	0.014
		<i>BasketballDrive</i>	0.590	0.008	0.037	0.266	0.004	0.008	0.488	0.006	0.043	0.206	0.003	0.015
		<i>RaceHorses</i>	0.462	0.009	0.029	0.263	0.006	0.009	0.486	0.008	0.029	0.228	0.003	0.015
	Class C	<i>BQMall</i>	0.881	0.012	0.017	0.586	0.007	0.002	0.807	0.008	0.020	0.369	0.003	0.009
		<i>PartyScene</i>	0.853	0.014	0.023	0.579	0.010	0.006	0.856	0.009	0.015	0.557	0.005	0.006
		<i>BasketballDrill</i>	0.620	0.008	0.027	0.240	0.003	0.007	0.753	0.011	0.021	0.422	0.005	0.011
		<i>RaceHorses</i>	0.694	0.013	0.012	0.490	0.011	0.001	0.731	0.012	0.008	0.420	0.006	0.004
	Class D	<i>BQSquare</i>	1.433	0.012	-0.002	0.966	0.008	0.000	1.379	0.009	0.002	1.147	0.007	-0.011
		<i>BlowingBubbles</i>	0.838	0.018	0.014	0.622	0.013	0.003	0.833	0.012	0.016	0.476	0.006	0.007
		<i>BasketballPass</i>	0.995	0.014	0.011	0.763	0.011	-0.003	0.947	0.011	0.007	0.585	0.006	0.003
		<i>FourPeople</i>	0.939	0.007	0.014	0.589	0.004	0.006	0.968	0.006	0.022	0.356	0.002	0.010
	Class E	<i>Johnny</i>	0.757	0.006	0.017	0.405	0.003	0.009	0.756	0.005	0.023	0.222	0.002	0.010
		<i>KristenAndSara</i>	0.908	0.006	0.020	0.508	0.003	0.008	0.829	0.005	0.025	0.299	0.002	0.010
		<b>Average</b>	<b>0.787</b>	<b>0.010</b>	<b>0.021</b>	<b>0.493</b>	<b>0.007</b>	<b>0.005</b>	<b>0.731</b>	<b>0.008</b>	<b>0.023</b>	<b>0.369</b>	<b>0.004</b>	<b>0.009</b>
QP 37 / CQ 55	<b>Average</b>		<b>0.697</b>	<b>0.014</b>	<b>0.013</b>	<b>0.413</b>	<b>0.007</b>	<b>0.003</b>	<b>0.675</b>	<b>0.010</b>	<b>0.012</b>	<b>0.354</b>	<b>0.005</b>	<b>0.006</b>
QP 42 / CQ 60	<b>Average</b>		<b>0.416</b>	<b>0.014</b>	<b>0.008</b>	<b>0.254</b>	<b>0.006</b>	<b>0.003</b>	<b>0.544</b>	<b>0.011</b>	<b>0.007</b>	<b>0.244</b>	<b>0.005</b>	<b>0.004</b>
QP 47 / CQ 63	<b>Average</b>		<b>0.221</b>	<b>0.011</b>	<b>0.004</b>	<b>0.112</b>	<b>0.005</b>	<b>0.005</b>	<b>0.394</b>	<b>0.011</b>	<b>0.004</b>	<b>0.145</b>	<b>0.005</b>	<b>0.004</b>

with QP 37, thus performing better on videos with quality levels closer to QP 42 and QP 47. However, it is evident that the BD-Rate value for STDF does not reflect the same quality in codecs other than HEVC, suggesting that the codec used in the training videos plays a significant role in the quality enhancement.

The lowest BD-Rate value for the MD-STDF models is -4.17% in AV1 with high QP/CQ. Although this is the lowest value, it still represents an improvement over the fixed quality points used in this study. In contrast, the standard STDF model shows a positive BD-Rate of 5.89% for AV1, indicating that, on average, the STDF model degrades the video quality compared to the videos decoded using only the codec. This demonstrates that MD-STDF provides a more consistent and generalized improvement in video quality across different codecs, outperforming the standard STDF model, which may not yield the same level of enhancement in all scenarios.

## 5 Visual Quality Perception

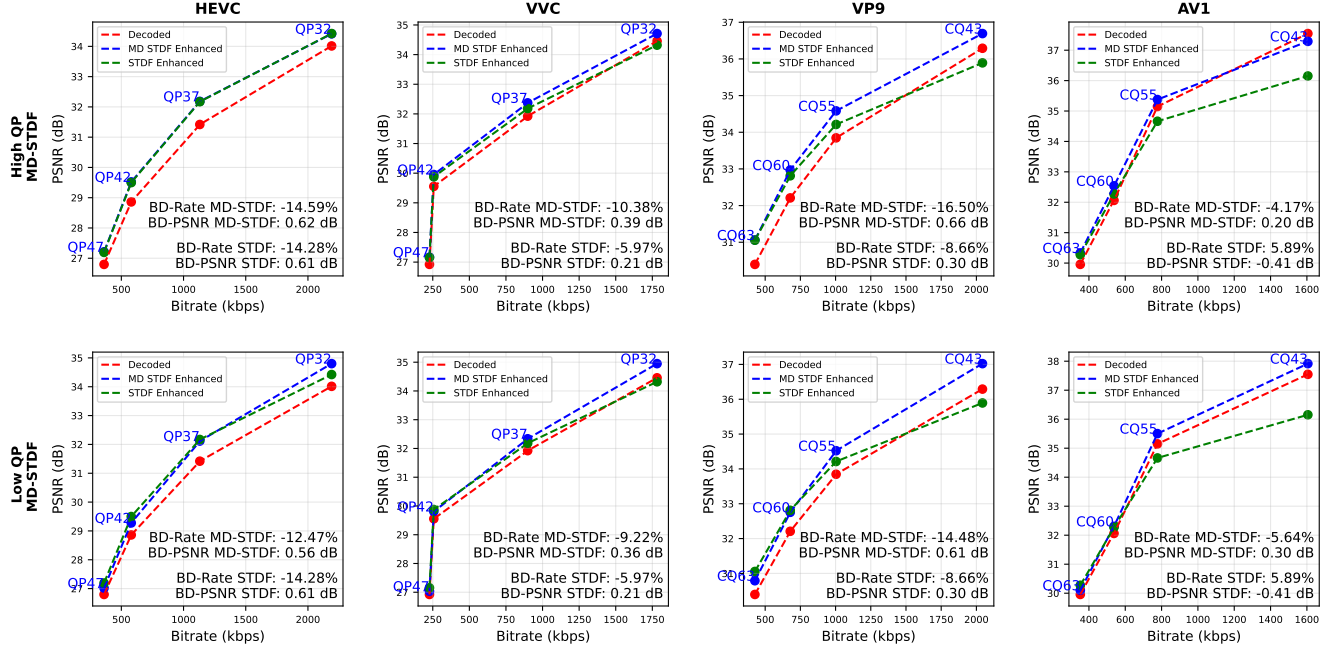
This section presents an analysis of the perceived visual quality improvement of the multi-domain STDF solution. The analysis was performed by the authors themselves. A medium resolution video (Class C) from the test set of all codecs was randomly selected, and from it, a frame with a high incidence of compression artifacts was chosen to provide clearer visual evidence for the article.

Figure 7 presents a composition based on frame number 14

of the *BasketballDrill* sequence, which was compressed with QP 37 (HEVC and VVC) and CQ 55 (VP9 and AV1). By observing the images in the second column, i.e., the images that went through the compression process, most of them exhibit a significant number of artifacts. The most deteriorated images are (c) and (i), which correspond to the HEVC and VP9 codecs, respectively. In image (l), which corresponds to the AV1 codec, some artifacts are still noticeable, although in smaller quantities. On the other hand, image (f), corresponding to the VVC codec, has the fewest artifacts.

In all the compressed images (second column), the blurring effect is noticeable, especially in the background, where fine details from the original image (first column) are lost due to the quantization process. The blocking effect, characterized by visible block boundaries and a lack of smooth transitions between adjacent blocks, can be observed mainly in images (c), (i), and (l). In image (c), this artifact creates a stair-step effect on the edges of the ball, distorting its circular shape. In image (i), it creates a mosaic effect, with blocky distortions spread throughout the frame. In image (l), the stair-step effect is also noticeable on the upper right edge of the ball, though less prominent than in (c).

By comparing the low-quality versions in the second column with their respective enhanced versions in the third column, it is evident that images (d) and (j) show a significant improvement in visual quality, with sharper edges and fewer visible artifacts. Image (m) also demonstrates a smoothing of artifacts, particularly in the background regions, leading to a more uniform appearance. Image (g), on the other hand, is



**Figure 6.** BD-Rate and BD-PSNR evaluation for High QP MD-STDF and Low QP MD-STDF, compared to the baseline STDF. The points on the chart indicate fixed quality values for each codec (HEVC and VVC: QP32, QP37, QP42, QP47; VP9 and AV1: QP43, QP55, QP60, QP63). Red curves: decoded videos without VQE; blue curves: decoded videos enhanced with the MD-STDF model; green curves: decoded videos enhanced with the baseline STDF.

the one that most resembles its compressed version before the application of the enhancement filter, as image (f) exhibits a low incidence of compression artifacts, making it challenging for the enhancement process to produce noticeable improvements. This highlights the limitations of enhancement techniques in scenarios where compression artifacts are minimal, as well as their effectiveness in restoring detail and reducing distortions in highly degraded frames.

## 6 Computational Cost Analysis

This section presents an analysis of the computational cost of the MD-STDF model, focusing on its execution time and computational cost per frame. All tests were conducted on the same machine used during the training phase. The evaluation was carried out using the HEVC test dataset with QP 37, and the execution time was measured for each frame individually.

Figure 8 a scatter plot illustrating the relationship between visual enhancement and computational cost. The x-axis represents the  $\Delta$ PSNR values, while the y-axis shows the number of Floating-point Operations (FLOPs), expressed in gigaflops (G). To calculate the FLOPs we choose the video *BQTerrace*, and used the same video in each model. The size of each circle reflects the number of parameters activated by each model during inference, and gray reference circles are included to serve as a visual legend for scale.

As observed, the MD-STDF model maintains the same number of parameters and FLOPs as the single-domain STDF-R3 model, while achieving a gain in  $\Delta$ PSNR. In contrast, architectures such as RF-DSTA [Zhao et al., 2021b] and CF-STIF [Luo et al., 2022]<sup>1</sup> igher improvements in

<sup>1</sup>The values of  $\Delta$  PSNR and number of parameters of RF-DSTA and

**Table 4.** Inference Time of VQE Architectures.

Model	Parameters	$\Delta$ PSNR	Class D	Frames Per Second (FPS)			
				Class C	Class E	Class B	Class A
STDF-R3	360414	0.756	114	40	20	7	4
RF-DSTA	1250230	0.910	64	18	7	2	1
CF-STIF	2220000	0.920	36	11	4	1	0.7
STDF-MC	360414	0.335	114	40	20	7	4
STDF-MD	360414	0.764	120	42	20	7	4

$\Delta$ PSNR but require significantly more parameters and computational resources per iteration.

In Table 4 we present the number of Frames Per Second (FPS) that each model can process, grouped by video resolution class. The MD-STDF model maintains nearly the same processing speed as the baseline STDF-R3 across all HEVC video resolutions, achieving up to 120.48 FPS in Class D and 3.83 FPS in Class A. It offers a slightly higher PSNR ( $\Delta$ PSNR of 0.764 vs. 0.756) while using the same number of parameters (360,414). In comparison, models like RF-DSTA and CF-STIF<sup>2</sup> provide higher visual quality improvements ( $\Delta$ PSNR of 0.910 and 0.920), but they require significantly more parameters and result in much lower frame rates (1.31 FPS and 0.73 FPS in Class A). These results show that MD-STDF offers a better balance between performance and computational cost, making it a suitable choice for real-time or resource-constrained applications.

## 7 Conclusion

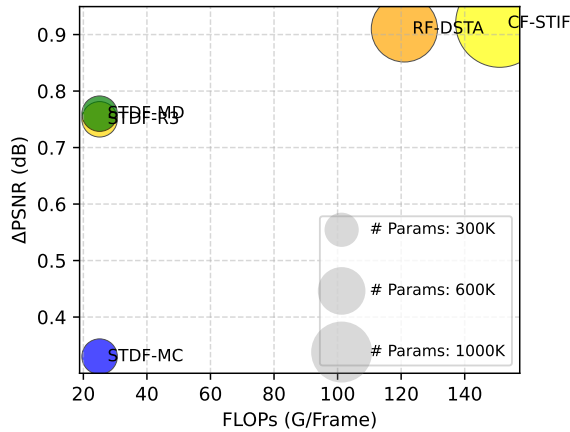
This work introduced the Multi-Domain Spatio-Temporal Deformable Fusion (MD-STDF) architecture, an innovative

CF-STIF are obtained direct from the papers of [Zhao et al., 2021b; Luo et al., 2022] with differently test conditions.

<sup>2</sup>For the inference time evaluation, a pre-trained model provided by the authors was used for RF-DSTA. For CF-STIF, since no pre-trained model was available, we trained it ourselves following the authors guidelines.



**Figure 7.** Frame number 14 of the *BasketballDrill* sequence: (a) Original frame (RAW); (b)(e)(h)(k) Cropped section of the original frame; (c) HEVC-decoded version; (d) Enhanced HEVC-decoded version; (f) VVC-decoded version; (g) Enhanced VVC-decoded version; (i) VP9-decoded version; (j) Enhanced VP9-decoded version; (l) AV1-decoded version; (m) Enhanced AV1-decoded version.



**Figure 8.** Computational cost analysis of MD-STDF over other VQE architectures.

model that employs a multi-domain learning approach to enhance the quality of compressed videos across various codecs, addressing limitations of previous works. The obtained results demonstrated the model's effectiveness, with consistent improvements in both objective and perceptual video quality, particularly in scenarios of severe compression.

The model demonstrated consistent gains across all evaluated metrics, including  $\Delta\text{PSNR}$ ,  $\Delta\text{SSIM}$ , and  $\Delta\text{LPIPS}$ . It achieved up to 0.764 dB improvement for HEVC videos at QP 37 and as much as 1.433 dB in specific sequences such as VQSquare. The model trained on low compression settings (low QP/CQ) achieved the best results across various codecs, while the high compression configuration showed robustness in more degraded conditions. BD-Rate analysis reinforced these findings, with significant improvements for VP9 (-16.50%) and HEVC (-14.59%). In contrast, the baseline STDF model not only underperformed but also degraded quality for AV1 (+5.89% BD-Rate), highlighting the supe-

rior generalization ability of MD-STDF. Importantly, these quality improvements were achieved without increasing the computational burden. The model retains the same number of parameters as STDF-R3 (360,414) and operates with similar computational cost, as demonstrated in the complexity analysis.

Finally, the results suggest that MD-STDF not only reduces compression artifacts but also provides perceptual visual enhancements, as evidenced by subjective analysis and complementary objective metrics. This performance highlights the potential of multi-domain learning in creating more generalized and effective models for video quality enhancement. This type of solution can be useful not only for improving the quality of compressed videos but also for enabling the use of more severe compression rates without negatively impacting the user experience, as a VQE model could be applied post-decoding.

## Declarations

## Acknowledgements

This study was financed 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).

## Authors' Contributions

**Garibaldi da Silveira Júnior:** Writing, Investigation, Experimental execution and Conceptualization.

**Gilberto Kreisler:** Writing, Investigation, Experimental execution and Conceptualization.



**Bruno Zatt:** Supervision, Writing - Review & Editing, Guidance in methodology and experimental design, Conceptual advice.

**Daniel Palomino:** Supervision, Writing - Review & Editing, Guidance in methodology and experimental design, Conceptual advice.

**Guilherme Corrêa:** Supervision, Writing - Review & Editing, Guidance in methodology and experimental design, Conceptual advice.

All authors contributed to the final manuscript, and all authors have read and approved the submitted version.

## Competing interests

The authors declare no competing interests relevant to the content of this article.

## Availability of data and materials

The code of our MD-STDF approach is available at <https://github.com/Espeto/md-stdf>.

The datasets and other softwares generated and/or analysed during the current study will be made upon request.

## References

- Agarwal, A., Agarwal, A., Sinha, S., Vatsa, M., and Singh, R. (2021). Md-csddnetwork: Multi-domain cross stitched network for deepfake detection. In *2021 16th IEEE international conference on automatic face and gesture recognition (FG 2021)*, pages 1–8. IEEE. DOI: 10.1109/FG52635.2021.9666937.
- Bender, I., Palomino, D., Agostini, L., Correa, G., and Porto, M. (2019). Compression efficiency and computational cost comparison between av1 and hevc encoders. In *2019 27th European Signal Processing Conference (EUSIPCO)*, pages 1–5. IEEE. DOI: 10.23919/EUSIPCO.2019.8903006.
- Bjontegaard, G. (2001). Calculation of average psnr differences between rd-curves. *ITU-T SG16 Q*, 6. Available at: [https://www.itu.int/wftp3/av-arch/video-site/0104\\_Aus/](https://www.itu.int/wftp3/av-arch/video-site/0104_Aus/). Document VCEG-M33.
- Boyce, J., Suehring, K., and Li, X. (2018). Jvet-j1010: Jvet common test conditions and software reference configurations. *JVET-J1010*. Available at: [https://www.researchgate.net/publication/326506581\\_JVET-J1010\\_JVET\\_common\\_test\\_conditions\\_and\\_software\\_reference\\_configurations](https://www.researchgate.net/publication/326506581_JVET-J1010_JVET_common_test_conditions_and_software_reference_configurations).
- Bross, B., Wang, Y.-K., Ye, Y., Liu, S., Chen, J., Sullivan, G. J., and Ohm, J.-R. (2021). Overview of the versatile video coding (vvc) standard and its applications. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(10):3736–3764. DOI: 10.1109/TCSVT.2021.3101953.
- Caballero, J., Ledig, C., Aitken, A., Acosta, A., Totz, J., Wang, Z., and Shi, W. (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*, pages 4778–4787. DOI: 10.1109/CVPR.2017.304.
- Chen, S. and Ye, M. (2021). Two-stage multi-frame cooperative quality enhancement on compressed video. In *2021 11th International Conference on Intelligent Control and Information Processing (ICICIP)*, pages 94–99. IEEE. DOI: 10.1109/icicip53388.2021.9642200.
- Chen, Y., Lu, R., Zou, Y., and Zhang, Y. (2018a). Branch-activated multi-domain convolutional neural network for visual tracking. *Journal of Shanghai Jiaotong University (Science)*, 23:360–367. DOI: 10.1007/s12204-018-1951-8.
- Chen, Y., Murherjee, D., Han, J., Grange, A., Xu, Y., Liu, Z., Parker, S., Chen, C., Su, H., Joshi, U., et al. (2018b). An overview of core coding tools in the av1 video codec. In *2018 picture coding symposium (PCS)*, pages 41–45. IEEE. DOI: 10.1109/PCS.2018.8456249.
- Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., and Wei, Y. (2017a). Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 764–773. DOI: 10.1109/ICCV.2017.89.
- Dai, Y., Liu, D., and Wu, F. (2017b). A convolutional neural network approach for post-processing in hevc intra coding. In *MultiMedia Modeling: 23rd International Conference, MMM 2017, Reykjavik, Iceland, January 4-6, 2017, Proceedings, Part I 23*, pages 28–39. Springer. DOI: 10.1007/978-3-319-51811-4\_3.
- Deng, J., Wang, L., Pu, S., and Zhuo, C. (2020). Spatio-temporal deformable convolution for compressed video quality enhancement. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 10696–10703. DOI: 10.1609/aaai.v34i07.6697.
- Dong, C., Deng, Y., Loy, C. C., and Tang, X. (2015). Compression artifacts reduction by a deep convolutional network. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 576–584. DOI: 10.1109/ICCV.2015.73.
- Fu, C.-M., Alshina, E., Alshin, A., Huang, Y.-W., Chen, C.-Y., Tsai, C.-Y., Hsu, C.-W., Lei, S.-M., Park, J.-H., and Han, W.-J. (2012). Sample adaptive offset in the hevc standard. *IEEE Transactions on Circuits and Systems for Video technology*, 22(12):1755–1764. DOI: 10.1109/TCSVT.2012.2221529.
- Guan, Z., Xing, Q., Xu, M., Yang, R., Liu, T., and Wang, Z. (2019). Mfqe 2.0: A new approach for multi-frame quality enhancement on compressed video. *IEEE transactions on pattern analysis and machine intelligence*, 43(3):949–963. DOI: 10.1109/TPAMI.2019.2944806.
- Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*. DOI: 10.48550/arXiv.1412.6980.
- Kreisler, G., Da Silveira, G., Zatt, B., Palomino, D., and Corrêa, G. (2024). Multi-codec video quality enhancement model based on spatio-temporal deformable fusion. In *2024 IEEE 15th Latin America Symposium on Circuits and Systems (LASCAS)*, pages 1–5. IEEE. DOI: 10.1109/LASCAS60203.2024.10506192.
- Kuanar, S., Conly, C., and Rao, K. (2018). Deep learning based hevc in-loop filtering for decoder quality enhancement. In *2018 Picture Coding Symposium (PCS)*, pages

- 164–168. IEEE. DOI: 10.1109/PCS.2018.8456278.
- Li, T., Xu, M., Zhu, C., Yang, R., Wang, Z., and Guan, Z. (2019). A deep learning approach for multi-frame in-loop filter of hevc. *IEEE Transactions on Image Processing*, 28(11):5663–5678. DOI: 10.1109/TIP.2019.2921877.
- Li, Z., Liu, F., Yang, W., Peng, S., and Zhou, J. (2021). A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE transactions on neural networks and learning systems*, 33(12):6999–7019. DOI: 10.1109/TNNLS.2021.3084827.
- Liang, M. and Hu, X. (2015). Recurrent convolutional neural network for object recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3367–3375. DOI: 10.1109/CVPR.2015.7298958.
- Luo, D., Ye, M., Li, S., and Li, X. (2022). Coarse-to-fine spatio-temporal information fusion for compressed video quality enhancement. *IEEE Signal Processing Letters*, 29:543–547. DOI: 10.1109/lsp.2022.3147441.
- Mac, K.-N. C., Joshi, D., Yeh, R. A., Xiong, J., Feris, R. S., and Do, M. N. (2019). Learning motion in feature space: Locally-consistent deformable convolution networks for fine-grained action detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6282–6291. DOI: 10.1109/ICCV.2019.00638.
- Meng, X., Deng, X., Zhu, S., and Zeng, B. (2019). Enhancing quality for vvc compressed videos by jointly exploiting spatial details and temporal structure. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 1193–1197. IEEE. DOI: 10.1109/ICIP.2019.8804469.
- Mukherjee, D., Bankoski, J., Grange, A., Han, J., Koleszar, J., Wilkins, P., Xu, Y., and Bultje, R. (2013). The latest open-source video codec vp9—an overview and preliminary results. In *2013 Picture Coding Symposium (PCS)*, pages 390–393. IEEE. DOI: 10.1109/PCS.2013.6737765.
- Nah, S., Son, S., and Lee, K. M. (2019). Recurrent neural networks with intra-frame iterations for video deblurring. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8102–8111. DOI: 10.1109/CVPR.2019.00829.
- Nam, H. and Han, B. (2016). Learning multi-domain convolutional neural networks for visual tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4293–4302. DOI: 10.1109/CVPR.2016.465.
- Norkin, A., Bjontegaard, G., Fuldseth, A., Narroschke, M., Ikeda, M., Andersson, K., Zhou, M., and Van der Auwera, G. (2012). Hevc deblocking filter. *IEEE Transactions on Circuits and Systems for Video Technology*, 22(12):1746–1754. DOI: 10.1109/TCSVT.2012.2223053.
- Peng, B., Chang, R., Pan, Z., Li, G., Ling, N., and Lei, J. (2022). Deep in-loop filtering via multi-domain correlation learning and partition constraint for multi-view video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(4):1911–1921. DOI: 10.1109/TCSVT.2022.3213515.
- Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer. DOI: 10.1007/978-3-319-24574-2\_28.
- Sandvine, I. (2023). Global internet phenomena report. *North America and Latin America*. Available at: <https://www.applogicnetworks.com/phenomena>.
- Sullivan, G. J., Ohm, J.-R., Han, W.-J., and Wiegand, T. (2012). Overview of the high efficiency video coding (hevc) standard. *IEEE Transactions on circuits and systems for video technology*, 22(12):1649–1668. DOI: 10.1109/TCSVT.2012.2221191.
- Tong, J., Wu, X., Ding, D., Zhu, Z., and Liu, Z. (2019). Learning-based multi-frame video quality enhancement. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 929–933. IEEE. DOI: 10.1109/ICIP.2019.8803786.
- Tsai, C.-Y., Chen, C.-Y., Yamakage, T., Chong, I. S., Huang, Y.-W., Fu, C.-M., Itoh, T., Watanabe, T., Chu-joh, T., Karczewicz, M., et al. (2013). Adaptive loop filtering for video coding. *IEEE Journal of Selected Topics in Signal Processing*, 7(6):934–945. DOI: 10.1109/JSTSP.2013.2271974.
- Wang, X., Chan, K. C., Yu, K., Dong, C., and Change Loy, C. (2019). Edvr: Video restoration with enhanced deformable convolutional networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 0–0. DOI: 10.1109/CVPRW.2019.00247.
- Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612. DOI: 10.1109/TIP.2003.819861.
- Xue, T., Chen, B., Wu, J., Wei, D., and Freeman, W. T. (2019). Video enhancement with task-oriented flow. *International Journal of Computer Vision*, 127:1106–1125. DOI: 10.1007/s11263-018-01144-2.
- Yang, R. and Timofte, R. (2021). NTIRE 2021 challenge on quality enhancement of compressed video: Dataset and study. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. DOI: 10.1109/CVPRW53098.2021.00076.
- Yang, R., Xu, M., Liu, T., Wang, Z., and Guan, Z. (2018a). Enhancing quality for hevc compressed videos. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(7):2039–2054. DOI: 10.1109/TCSVT.2018.2867568.
- Yang, R., Xu, M., Wang, Z., and Li, T. (2018b). Multi-frame quality enhancement for compressed video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6664–6673. DOI: 10.1109/CVPR.2018.00697.
- Zhang, R., Isola, P., Efros, A. A., Shechtman, E., and Wang, O. (2018). The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595. DOI: 10.48550/arXiv.1801.03924.
- Zhao, H., Zheng, B., Yuan, S., Zhang, H., Yan, C., Li, L., and Slabaugh, G. (2021a). Cbren: Convolutional neural

- networks for constant bit rate video quality enhancement. *IEEE Transactions on Circuits and Systems for Video Technology*. DOI: 10.1109/TCSVT.2021.3123621.
- Zhao, M., Xu, Y., and Zhou, S. (2021b). Recursive fusion and deformable spatiotemporal attention for video compression artifact reduction. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 5646–5654. DOI: 10.1145/3474085.3475710.
- Zhu, C., Dong, H., Pan, J., Liang, B., Huang, Y., Fu, L., and Wang, F. (2022). Deep recurrent neural network with multi-scale bi-directional propagation for video deblurring. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 3598–3607. DOI: 10.1609/aaai.v36i3.20272.
- Zhu, X., Hu, H., Lin, S., and Dai, J. (2019). Deformable convnets v2: More deformable, better results. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9308–9316. DOI: 10.1109/CVPR.2019.00953.