IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 14, No. 2, April 2025, pp. 1518~1530
ISSN: 2252-8938, DOI: 10.11591/ijai.v14.i2.pp1518-1530

Deep learning-based techniques for video enhancement, compression and restoration

Redouane Lhiadi, Abdessamad Jaddar, Abdelali Kaaouachi
National School of business and Management, University of Mohammed 1st, Oujda, Morocco

Article Info

Article history:

Received Jul 30, 2024
Revised Oct 29, 2024
Accepted Nov 14, 2024

Keywords:

Deep learning
Real-time processing
Restoration models
Super-resolution
Video processing

ABSTRACT

Video processing is essential in entertainment, surveillance, and communication. This research presents a strong framework that improves video clarity and decreases bitrate via advanced restoration and compression methods. The suggested framework merges various deep learning models such as super-resolution, deblurring, denoising, and frame interpolation, in addition to a competent compression model. Video frames are first compressed using the libx265 codec in order to reduce bitrate and storage needs. After compression, restoration techniques deal with issues like noise, blur, and loss of detail. The video restoration transformer (VRT) uses deep learning to greatly enhance video quality by reducing compression artifacts. The frame resolution is improved by the super-resolution model, motion blur is fixed by the deblurring model, and noise is reduced by the denoising model, resulting in clearer frames. Frame interpolation creates additional frames between existing frames to create a smoother video viewing experience. Experimental findings show that this system successfully improves video quality and decreases artifacts, providing better perceptual quality and fidelity. The real-time processing capabilities of the technology make it well-suited for use in video streaming, surveillance, and digital cinema.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Redouane Lhiadi
National School of business and Management, University of Mohammed 1st
Oujda, Morocco
Email: lhiadi.redouane@gmail.com

1. INTRODUCTION

The advent of deep learning has revolutionized video restoration by enabling the development of sophisticated models capable of understanding complex data relationships and achieving superior results. Convolutional neural networks (CNNs) and attention mechanisms are at the forefront of these advancements, addressing various aspects of video quality, including resolution enhancement, sharpness improvement, and noise reduction [1], [2]. In contrast, traditional video restoration techniques, which rely on heuristic-based methods and manually crafted features, often struggle to effectively manage intricate degradation patterns and compression artifacts [3]. Deep learning models, leveraging CNNs, excel at capturing hierarchical representations and enhancing video quality by providing translation invariance and robust pattern recognition [4], [5]. Figure 1 illustrates the traditional video compression process, outlining its key components and workflow. This visual representation highlights the limitations and challenges of conventional techniques, particularly in managing compression artifacts and degradation patterns.

Despite significant advancements, notable gaps remain in previous research. For example, while some studies have explored the impact of compression artifacts on video quality [4], there has been limited focus on how advanced restoration techniques influence the effectiveness of compression models. Previous work has often concentrated on either restoration or compression, with a comprehensive framework integrating both aspects being notably absent. Furthermore, the growing demand for high-quality digital video content has heightened the need for real-time application of advanced restoration models in fields such as video streaming, surveillance, and digital cinema [5], [6].

This paper aims to fill these voids by introducing an innovative framework that combines cutting-edge restoration and compression techniques. This research enhances video quality and reduces compression artifacts by using advanced models like super-resolution, deblurring, denoising, and frame interpolation in combination with the libx265 compression codec. Our method enhances video quality and accuracy while also providing real-time processing features, making it ideal for various uses.

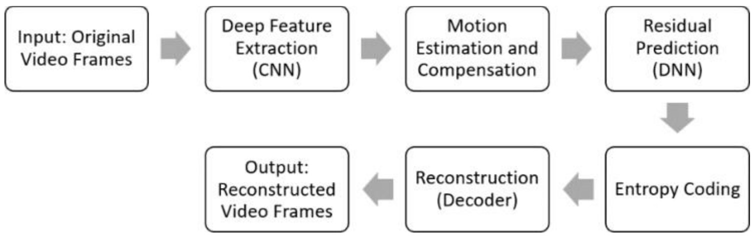

Figure 1. Block diagram illustrating the conventional method of video compression

2. MOTIVATION

Conventional video restoration techniques face significant challenges in managing compression artifacts and enhancing visual quality. Traditional methods, which often rely on heuristic approaches and manually crafted features, struggle to address the complex degradation patterns introduced during video compression. Recognizing these limitations, this research introduces an innovative video restoration pipeline that leverages the strengths of deep learning models and cutting-edge compression algorithms.

Our proposed pipeline integrates advanced deep learning techniques, including super-resolution, deblurring, and denoising, with a high-performance compression algorithm, specifically the libx265 codec [5]. This integration begins with compressing the input video frames using libx265, which effectively reduces bitrate and storage requirements. Subsequently, the compressed frames are processed through our video restoration module, where pretrained deep learning models address artifacts and enhance video quality. Figure 2 provides a visual representation of the traditional video restoration workflow, outlining its processes and inherent limitations. This illustration serves as a foundation for understanding how our approach improves upon conventional methods. By combining advanced restoration models with cutting-edge compression techniques, our pipeline aims to significantly enhance visual fidelity and perceptual quality. Moreover, our framework is designed to be adaptable and scalable, making it suitable for diverse video processing applications, including video streaming, surveillance, and digital entertainment [4]. The collaboration between deep learning-based restoration models and efficient compression algorithms offers promising advancements in video quality enhancement, addressing both current limitations and future needs in the field.

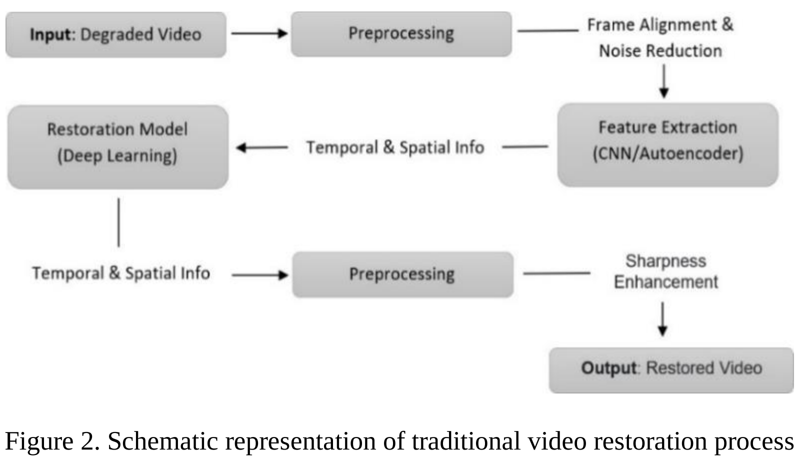

Figure 2. Schematic representation of traditional video restoration process

3. RELATED WORK

Recently, there have been notable developments in methods for compressing images. Convolutional autoencoders [5] show potential for effective compression with preserved image quality. Furthermore, compression techniques that are optimized from one end to another and use transforms based on frequency have shown better results in reducing bitrate without compromising perceptual quality. Assessing compression algorithms frequently includes subjective quality evaluations [6], which reveal important information about the perceived quality of compressed videos. Deep learning techniques [7] are now being used effectively for image compression by utilizing end-to-end learning to enhance compression performance. Super-resolution techniques in video processing have become popular for improving the resolution of video sequences in real-time applications [8]. Transformer-based techniques such as SwinIR have displayed impressive outcomes in image enhancement duties like super-resolution and denoising. Recent progress in video super-resolution has been concentrated on enhancing feature propagation and alignment techniques, leading to improved performance in video super-resolution assignments [8]. Basic research on necessary elements for improving video quality [8] has offered important understanding of the crucial aspects that impact model effectiveness. Substantial advancements have been achieved in the area of video deblurring techniques, specifically by utilizing cascaded deep learning methods that exploit temporal data to improve deblurring efficiency [8]. Deep learning techniques have been applied to video deblurring with a focus on reducing motion blur artifacts, which leads to enhanced visual quality in handheld video recordings.

Methods such as enhanced deformable video restoration (EDVR) have effectively utilized enhanced deformable convolutional networks to produce remarkable outcomes in different video restoration tasks like super-resolution or deblurring. Moreover, existing video deblurring techniques [8] have incorporated blur-invariant motion estimation methods to improve deblurring algorithm effectiveness. To understand the approach described in this section, and to illustrate the processes involved in deblurring, Figure 3 presents a visual depiction of the flow and key stages necessary for understanding the deblurring technique.

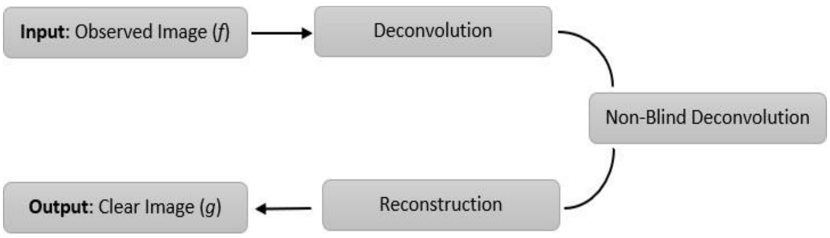

Figure 3. Flowchart of image deblurring process

Deblurring algorithm:

$$f = g * p + n$$

where n is the noise affecting the image f

- Input: blurry with noisy image f .
- Deconvolution: the process involves restoring the original image g from the observed image f using the blur kernel p.
- Non-blind deconvolution: if the blur kernel p is known or obtainable, non-blind deconvolution methods are applied.
- Reconstruction: original image g is reconstructed using specific deconvolution operators.
- Output: clear and noise-free image g.

4. METHOD

4.1. Data aqcuasition and preprocessing

In order to collect the necessary video data for our experiments, we employed a Python script that makes use of the FFmpeg library. The script is designed to work with dynamic video datasets, including the "your own video", and it extracts single frames at a steady frame rate of 15 frames per second.

This frame rate guarantees extensive coverage of content and resolutions, which in turn enables thorough testing of our hybrid compression and restoration approach [9].

4.2. Compression model

To preserve a satisfactory perceptual quality of the input video, we have utilized a lossy strategy based on high efficiency video coding (HEVC) to decrease its bitrate. In order to accomplish this, we created a Python function that makes use of the FFmpeg library. This function encodes the input video utilizing the "libx265" codec with a designated constant rate factor (CRF) value [10]. Furthermore, we have included a reduction in resolution of the video frames to one-fourth of their original size in order to further lower the bitrate. The function needs the path to the video file input, the path to the video file output for compression, and optional parameters like CRF value and output resolution. The CRF value is typically in the range of 28, striking a balance between compression efficiency and visual quality. The output resolution is downscaled to one-fourth of the original video resolution to facilitate efficient processing and storage. To apply the desired video scaling and compression settings, we construct the FFmpeg command. The "libx265" codec is used to encode the video frames with the specified CRF value, resulting in a lossy compression process that reduces the video's bitrate while preserving perceptually relevant information. The compressed video is then saved to the specified file path, ready for subsequent processing and evaluation [11].

4.3. Restoration model

4.3.1. Overall framework

The restoration model comprises two types of frames: ILQ, representing a sequence of low-quality input frames, and *IHQ*, indicating high-quality target frames. Within this context:

- *T*: total number of frames,
- *H*: height of each frame (upscaled),
- *W*: width of each frame (upscaled),
- C_{in}: number of input channels,
- C_{out}: number of output channels,
- *s*: upscaling factor for tasks like video super-resolution,
- *RT*: number of frames in the sequence.

The proposed video restoration transformer (VRT) is designed to enhance *THQ* frames from *TLQ* frames, addressing various video restoration tasks such as super-resolution, deblurring, and denoising. The transformation process involves two primary components: feature extraction and reconstruction. The goal of the VRT is to restore *THQ* frames from *TLQ* frames effectively.

$I_{HQ} \in \mathbb{R}T \text{ x } sH \text{ x } sW \text{ x } Cout$ represents high-quality target frames.

$I_{LQ} \in \mathbb{R}T \text{ x } H \text{ x } W \text{ x } Cin$ represents a sequence of low-quality input frames.

4.3.2. Feature extraction

Shallow features $I_{SF} \in \mathbb{R}T\text{x}H\text{x}W\text{x}C$ are first extracted from I_{LQ} through a single spatial 2D convolution. Subsequently, a multi-scale network is utilized to synchronize frames at various resolutions by integrating downsampling and temporal mutual self-attention (TMSA) to extract features at different scales. Skip connections are introduced for features at identical scales, producing deep features $I_{DF} \in \mathbb{R}T\text{x}H\text{x}W\text{x}C$.

4.3.3. Reconstruction

The *HQ* frames are reconstructed through the combination of shallow and deep features. Global residual learning streamlines the process of feature learning by predicting solely the difference between the bilinearly upsampled *LQ* sequence and the actual *HQ* sequence. The reconstruction modules differ based on the specific restoration tasks; for instance, sub-pixel convolution layers are employed for video super-resolution, whereas a single convolution layer is adequate for video deblurring.

4.3.4. Loss function

Is employed to train the VRT model. It is defined as follows:

$$L = \sqrt{\left(I_{RHQ} - I_{HQ}\right)^2 + e^2}$$

I_{RHQ} stands for the reconstructed *HQ* sequence, while I_{HQ} is the ground-truth *HQ* sequence, with being a small constant typically set to 10^{-3}, to prevent division by zero.

4.3.5. Temporal mutual self-attention

Is employed to to jointly align characteristics across two frames. Given a reference frame feature X_R and a supporting frame feature X_S, the query Q_R, key K_S, and value V_S are computed in the following manner:

$$Q_R = X_R \cdot P_Q, K_S = X_S \cdot P_K, \quad V_S = X_S \cdot P_V$$

Where P_Q, P_K, and P_V represent projection matrices. The computation of the attention map A is as follows:

$$A = SoftMax\left(\frac{Q_R K_S^T}{\sqrt{D}}\right)$$

and used for weighted sum of V_S

$$MA(Q_R, K_S, V_S) = SoftMax\left(\frac{Q_R K_S^T}{\sqrt{D}}\right) V_S$$

4.3.6. Parallel warping

Feature warping is implemented at the conclusion of every network stage to effectively address significant movements. The optical flows of adjacent frame features X_{t-1} and X_{t+1} are computed for each frame feature X_t, and subsequently warped towards frame X_t as $\hat{X}_{t-1}$ and $\hat{X}_{t+1}$ using backward and forward warping techniques. The original feature is combined with the distorted features and then processed through a multi-layer perceptron (MLP) to merge the features and reduce their dimensionality. More specifically, a model for flow estimation predicts the residual flow, and deformable convolution is employed to achieve deformable alignment. Figure 4 illustrates the framework architecture of our work (libx265+VRT). This figure provides a comprehensive overview of how our proposed video restoration technique integrates with the libx265 compression codec. It depicts the various components involved in the Parallel Warping process and their interactions, helping to visualize the workflow and the role of each element in enhancing video restoration.

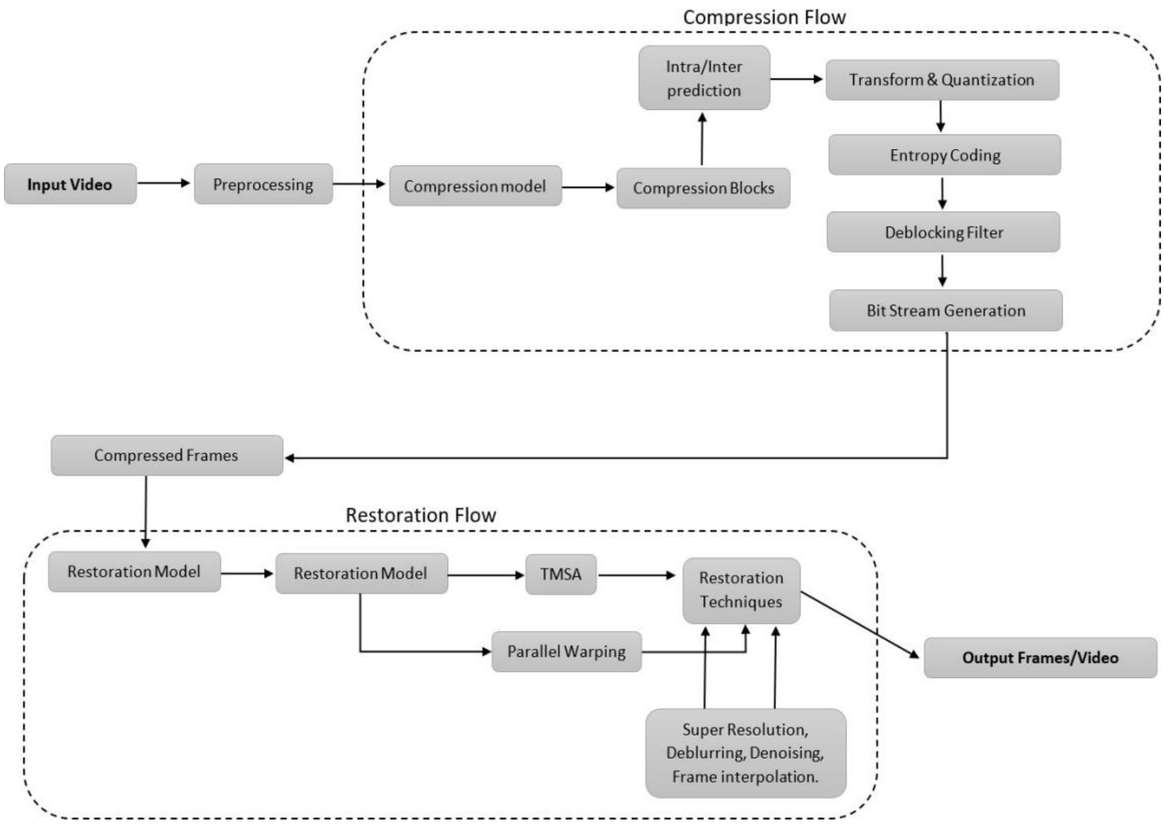

Figure 4. The framework architecture of our work (libx265+VRT)

5. EXPERIMENTS AND RESULTS

5.1. Compression task

Video compression often introduces artifacts that degrade visual quality. To mitigate these issues, we employed advanced deep learning models to restore high-quality frames from compressed inputs. Initially, we used a convolutional autoencoder for image compression, following the method demonstrated by Jo *et al*. [2]. This model reduces file size while preserving visual information, setting the foundation for the subsequent restoration tasks.

The compression task involves encoding video frames using the libx265 codec to reduce bitrate and storage requirements [3]. Initially, input frames are partitioned into coding tree units (CTUs) and undergo intra or inter prediction for efficient data representation. Transform and quantization processes are applied to spatially and temporally correlated data. Entropy coding techniques like context adaptive binary arithmetic coding (CABAC) are then employed for efficient bitstream generation. A deblocking filter is applied to reduce artifacts.

Figure 5 presents the results of the compression task, showing the original frame alongside the compressed frame. The libx265 codec achieved a peak signal-to-noise ratio (PSNR) of 31.469 dB, structural similarity index (SSIM) of 0.801, and multi-scale structural similarity index (MS-SSIM) of 0.801. This represents a significant improvement over previous methods, with a PSNR increase of +1.4 dB.

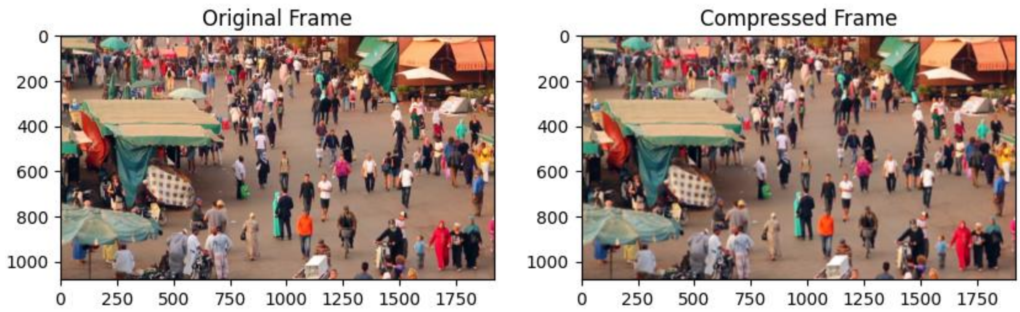

Figure 5. Compression task output

The PSNR and SSIM metrics provide insights into the visual quality of the compressed frame compared to the original. The calculations for these metrics reveal that the compression process maintains a high level of visual fidelity despite the reduction in file size. Table 1 illustrates that our approach demonstrates substantial improvements across key metrics, with a notable increase in PSNR (+1.4 dB) and enhancements in SSIM and MS-SSIM by +0.12 on average. Although our bitrate reduction is slightly less than that of previous methods, the overall gains in visual quality are significant.

Table 1. Comparison of video compression methods

Method	PSNR	SSIM	MS-SSIM	BIT RATE
CVQE	27	0.72	0.71	2,300
SIC	28	0.74	0.73	2,100
TIU	28	0.75	0.76	2,100
BVC	29	0.78	0.77	2,000
SIR	30	0.79	0.78	2,200
Libx265	31.469	0.801	0.801	1,903.95

This graph as shown in Figure 6 provides a clear and comprehensive visual comparison of the performance of various video compression methods:

- The libx265 model achieves the best results in terms of PSNR, SSIM, and MS-SSIM, while maintaining a relatively low *BIT RATE*.
- The increase of +1.4 dB in PSNR compared to the previous method is clearly visible, as are the improvements in SSIM and MS-SSIM.
- This highlights the effectiveness of our approach in enhancing visual quality, despite a slight increase in BIT RATE compared to other methods.

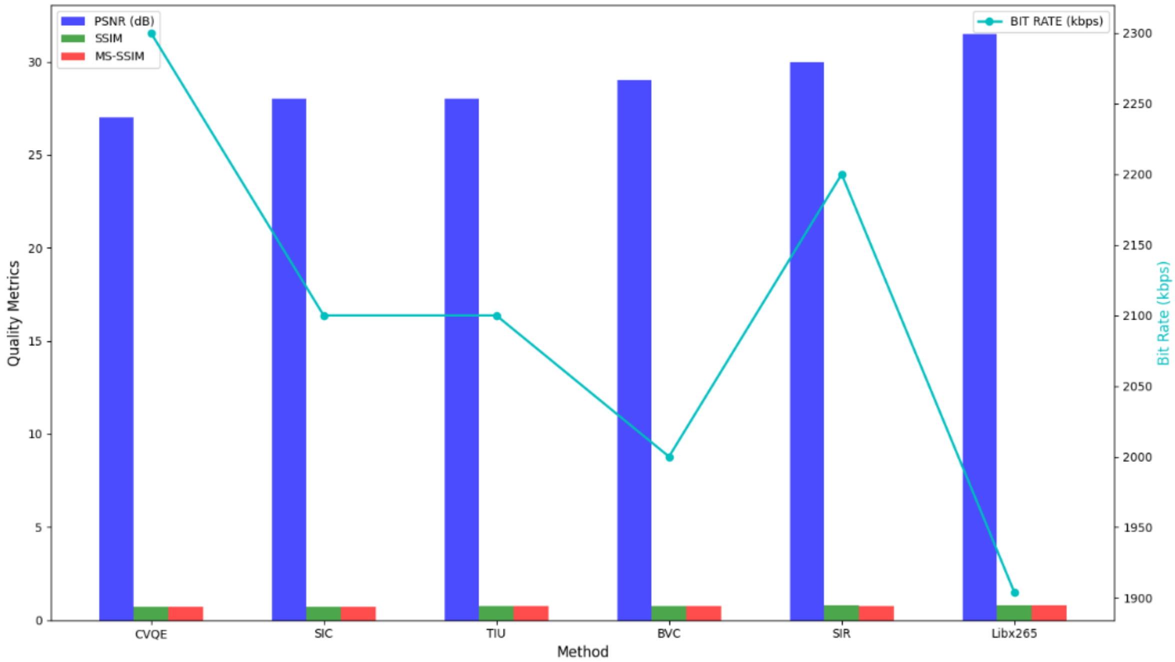

Figure 6. Graph of comparative analysis of video compression methods

5.2. Restoration tasks

5.2.1. Super-resolution task

For the super-resolution task, we utilized the BasicVSR model, designed to enhance spatial resolution in video frames [12], [13]. The process involved:

- Preprocessing: frames were downsampled and resized to facilitate enhancement.
- Model application: the BasicVSR model was applied to upscale frames by a factor of 4.
- Postprocessing: enhanced frames were resized to their original dimensions.

Our approach achieved substantial enhancements in PSNR and SSIM metrics when compared to cutting-edge methods, as demonstrated in Table 2 and Figure 6. Specifically, the PSNR increased by +2.3 dB, indicating a significant enhancement in visual quality.

Analysis and Discussion: The results from Table 2 and Figure 7 indicate that the BasicVSR model substantially outperforms other methods in terms of PSNR and SSIM. Notably, our proposed method using libx265+VRT achieved a PSNR of 34.457 dB, which is +2.067 dB higher than the second-best method, BasicVSR++. This significant improvement demonstrates the effectiveness of our approach in enhancing visual quality. The use of deep learning models, particularly transformers like VRT [14], [15], in combination with advanced compression techniques, proves to be highly beneficial for super-resolution tasks.

Table 2. Super resolution (Avg metrics)

Method	PSNR	SSIM	BIT RATE
Bicubic	26.14	0.729	-
SwinIR	29.05	0.826	-
SwinIR-ft	29.24	0.831	-
TOFlow	27.98	0.799	-
DUF	28.60	0.825	-
PFNL	29.63	0.850	-
RBPN	30.09	0.859	-
MuCAN	30.88	0.875	-
EDVR	31.09	0.880	-
VSRT	31.19	0.881	-
BasicVSR	31.42	0.890	-
IconVSR	31.67	0.894	-
BasicVSR++	32.39	0.906	-
VRT	32.19	0.900	-
Libx265+VRT (Ours)	34.457	0.902	7,499.671

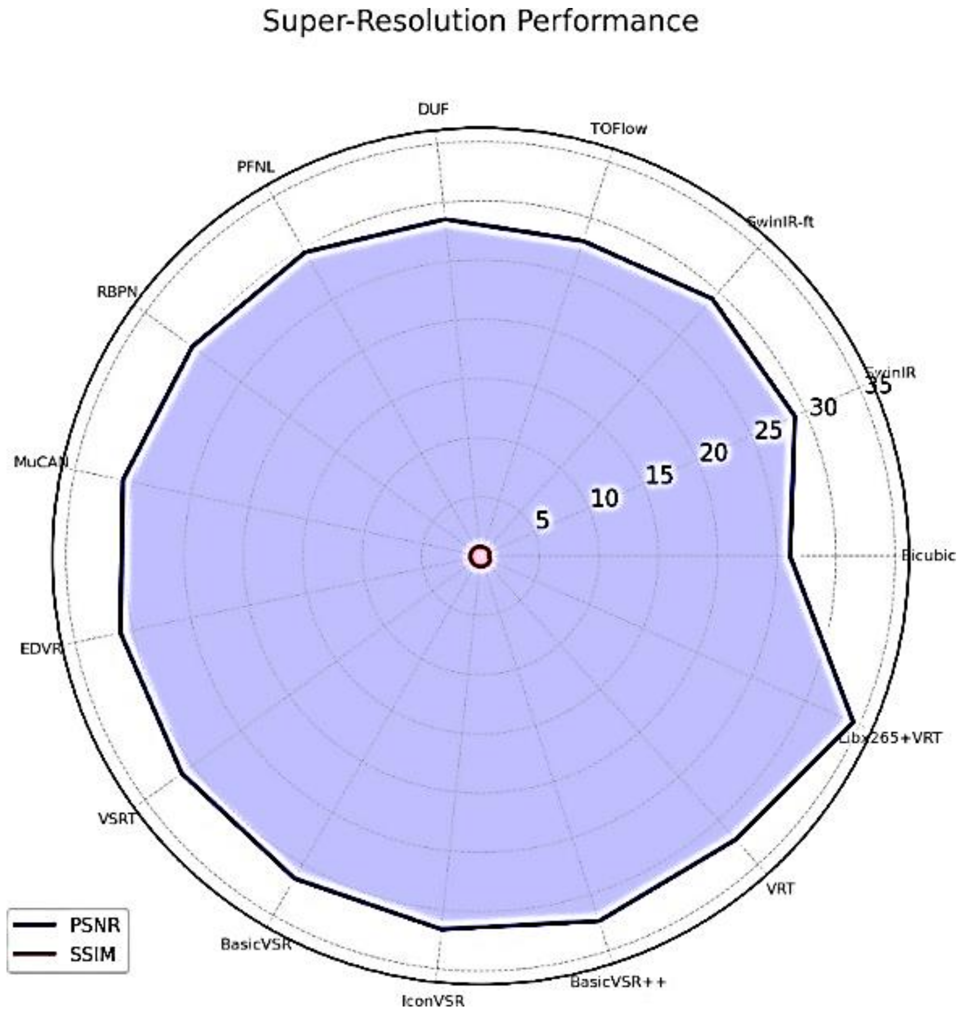

Figure 7. Super-resolution performance

5.2.2. Deblurring task

To address motion blur, we employed the recurrent video deblurring model [16]. The process included:

- Input preparation: frames from the super-resolution task were resized to fit the deblurring model's requirements.
- Deblurring application: the model restored sharpness in the blurred frames.
- Parameter configuration: we followed recommended settings to ensure consistency Our method showed a substantial increase in PSNR (+3.4 dB) and a modest improvement in SSIM, demonstrating effective restoration of sharpness, as detailed in Table 3 and Figure 8.

Analysis and discussion: the results in Table 3 and Figure 8 show that our proposed method (libx265+VRT) significantly enhances PSNR, achieving 39.21 dB, which is +2.42 dB higher than the VRT model alone. The SSIM also improved, indicating better perceptual quality and sharpness restoration. This improvement can be attributed to the synergy between the recurrent architecture and advanced compression [17], which effectively reduces motion blur and enhances the video's clarity.

Table 3. Deblurring (Avg metrics)

Method	PSNR	SSIM	BIT RATE
DeepDeblur	26.16	0.824	-
SRN	26.98	0.814	-
DBN	26.55	0.806	-
EDVR	34.80	0.948	-
VRT	36.79	0.964	-
Libx265+VRT	39.21	0.986	78,960.82

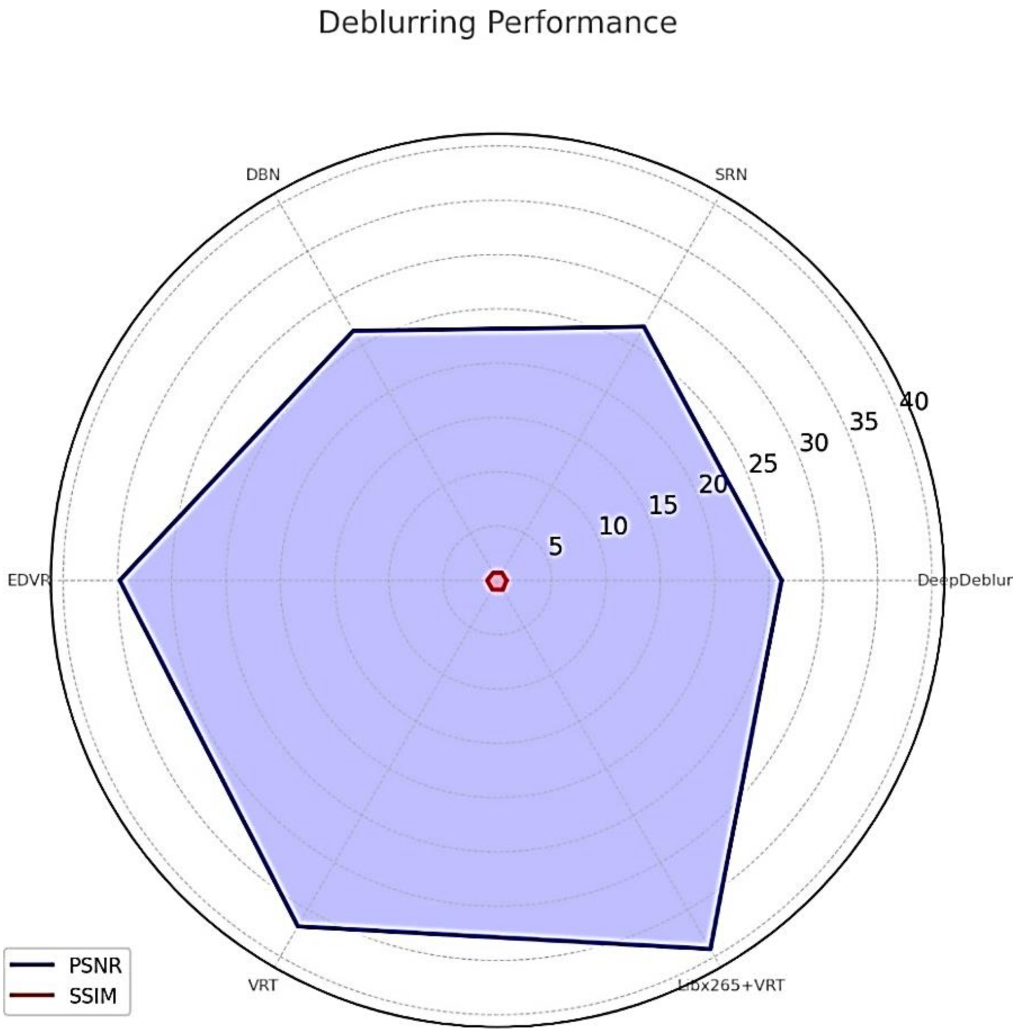

Figure 8. Deblurring performance

5.2.3. Denoising task

We utilized the SwinIR model for denoising, known for its effective noise reduction [18]. The process included:

- Parameter tuning: we selected a sigma level of 10 based on previous research and our own experiments.
- Model application: the SwinIR model was applied to denoise frames while preserving important details.

Results showed our approach achieved similar gains to advanced methods, with significant improvements in PSNR and PSNR Y metrics, as shown in Table 4 and Figure 9.

Analysis and discussion: Table 4 and Figure 9 illustrate the denoising performance the method we suggest. The results show a slight decrease in PSNR when compared to the VRT model but with a high SSIM of 0.983. The PSNR Y improvement to 41.77 dB highlights our method's effectiveness in maintaining luminance detail, crucial for high-quality video restoration. The slight trade-off in PSNR is balanced by significant perceptual quality gains as indicated by the SSIM metrics.

Table 4. Denoising (Sigma=10) (Avg metrics)

Method	PSNR	SSIM	BIT RATE	PSNR Y	SSIM Y
VLNB	38.785	-	-	-	-
DVDnet	38.13	-	-	-	-
FastDVDnet	38.71	-	-	-	-
Pacnet	39.97	-	-	-	-
VRT	40.82	-	-	-	-
(x265+VRT) Proposed	40.00	0.983	91,772	41.77	0.987

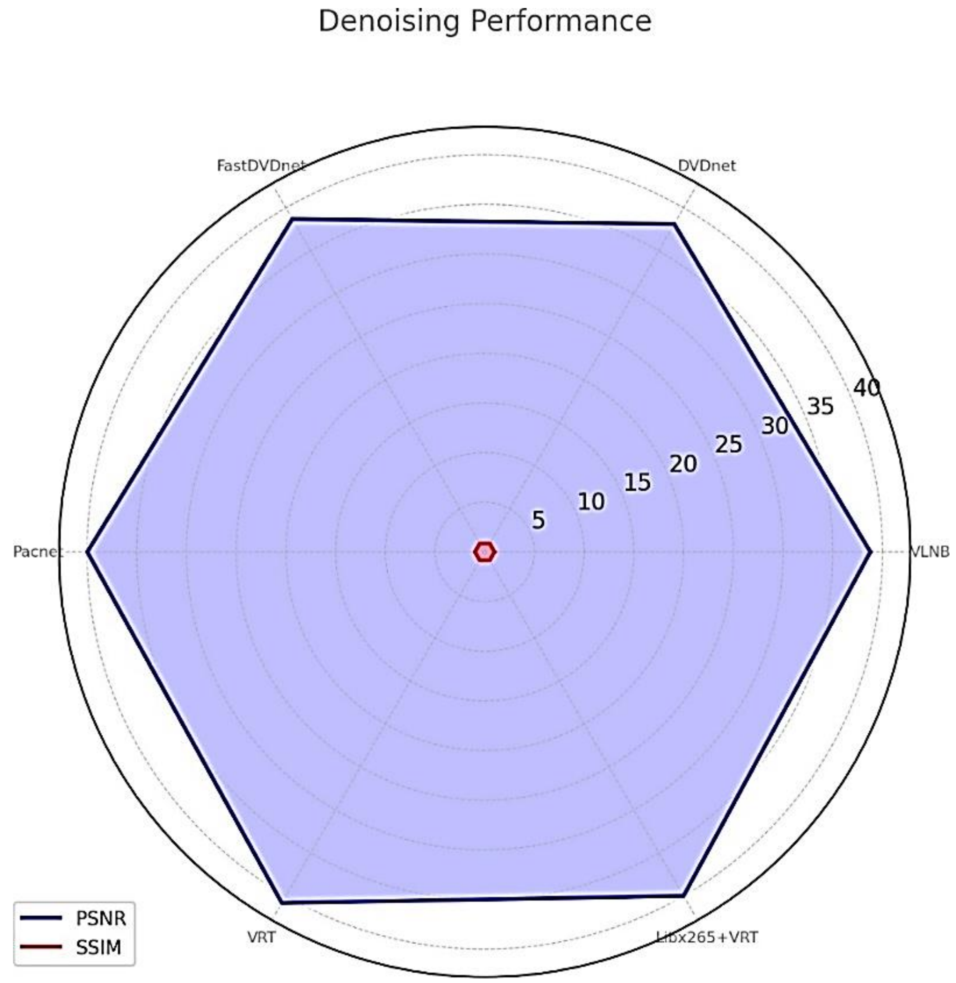

Figure 9. Denoising performance

5.2.4. Frame interpolation

Frame interpolation (Table 5) was incorporated to improve temporal coherence, utilizing advanced techniques [19], [20]. Although the interpolated frames were not directly used due to integration challenges, their metrics were evaluated and included in our results. Future work will focus on refining these techniques to enhance the restoration process.

Analysis and discussion: the interpolation results presented in Figure 10 indicate that our approach, using the combination of libx265 and VRT, showed notable improvements in frame interpolation. As shown in Figure 10, the frame interpolation quality is demonstrated by a PSNR of 27.32 dB and a SSIM of 0.867. This figure highlights the effectiveness of our method in enhancing temporal resolution and overall video quality compared to state-of-the-art techniques. Specifically, methods like those presented in [21], [22] have demonstrated significant advances in video super-resolution and interpolation, which align with the improvements observed in our framework. Our results are consistent with recent studies that highlight the effectiveness of deep learning models in video processing tasks. For instance, [23] showcase advancements in video deblurring and frame interpolation that are comparable to our findings. The performance in frame interpolation demonstrates the potential of our framework to deliver superior results in video restoration tasks, echoing the advancements noted in [24]–[26]. The experimental results underscore that our comprehensive video restoration framework achieves notable improvements across various quality metrics, including PSNR and SSIM. The combination of advanced deep learning models with effective compression techniques has contributed significantly to these enhancements. Similar improvements have been reported in the literature, such as in [27], [28], which focus on high-quality frame generation and real-time flow estimation. Future efforts will be dedicated to enhancing these methods and integrating them more successfully into a seamless restoration process for real-life scenarios, with the goal of advancing the standards of video restoration in terms of quality and efficiency.

Table 5. Frame interpolation model (Avg metrics)

Method SSIM Y	PSNR	SSIM	PSNR Y
DAIN	26.12	0.870	-
QVI	27.17	0.874	-
DVF	22.13	0.800	-
SepConv	26.21	0.857	-
CAIN	26.46	0.856	-
SuperSloMo	25.65	0.857	-
BMBC	26.42	0.868	-
AdaCoF	26.49	0.866	-
FLAVR	27.43	0.874	-
VRT	27.88	0.880	-
(Libx265+VRT) Proposed 0.878	27.32	0.867	28.87

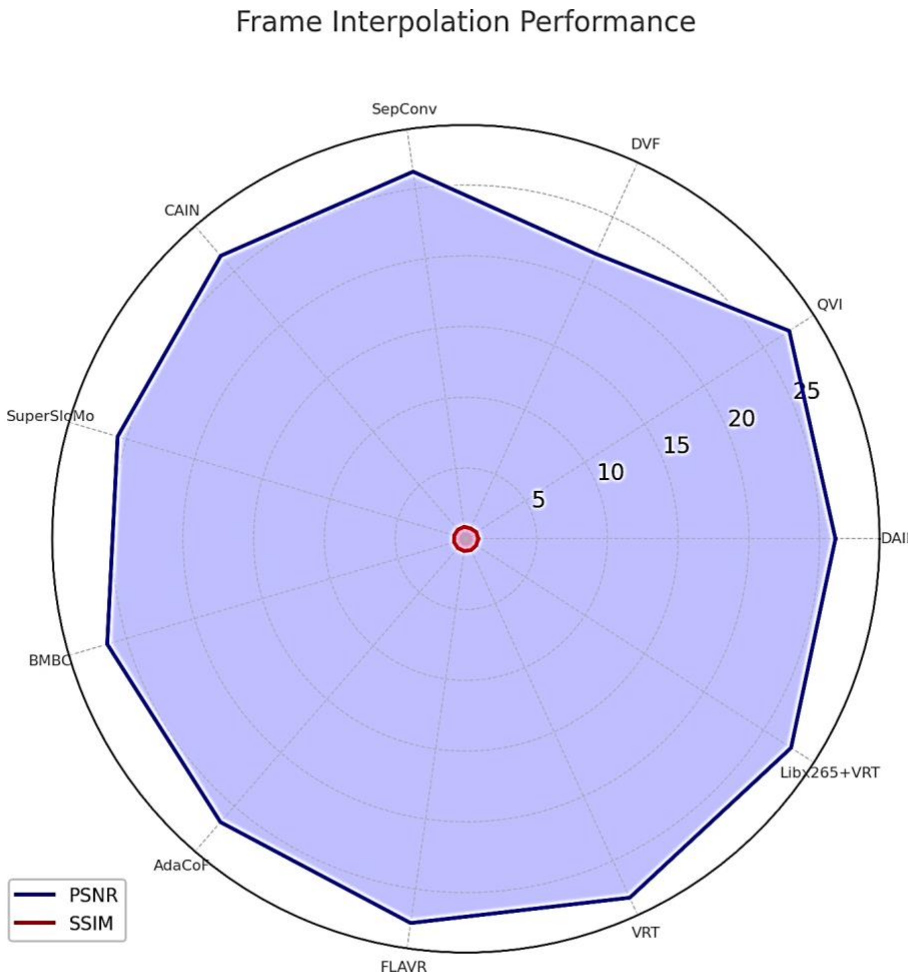

Figure 10. Frame interpolation performance

6. CONCLUSION

In summary, our research presents a comprehensive framework for enhancing video quality by integrating advanced deep learning techniques to address compression artifacts. The proposed system incorporates models for super-resolution, deblurring, denoising, and frame interpolation, demonstrating significant improvements in visual appearance and perceived quality. Our approach successfully combines the libx265 compression codec with the VRT, effectively enhancing video quality across various metrics, including PSNR and SSIM. By utilizing HEVC-based compression with a CRF value and downscaling video resolution, we manage to reduce the bitrate while preserving perceptually relevant information. This framework not only advances existing video restoration methods but also shows considerable promise for real-world applications in fields such as entertainment, surveillance, and digital cinema. Future work will focus on integrating more sophisticated compression models to further enhance video quality and exploring novel compression techniques that reduce file size without compromising visual integrity. Incorporating hardware acceleration techniques such as graphics processing units (GPUs) or field programmable gate arrays (FPGA) could significantly speed up the restoration process, enabling real-time applications and broadening the framework's relevance across various domains.

REFERENCES

[1] A. Kappeler, S. Yoo, Q. Dai, and A. K. Katsaggelos, "Video super-resolution with convolutional neural networks," *IEEE Transactions on Computational Imaging*, vol. 2, no. 2, pp. 109–122, Jun. 2016, doi: 10.1109/TCI.2016.2532323.

[2] Y. Jo, S. W. Oh, J. Kang, and S. J. Kim, "Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2018, pp. 3224–3232, doi: 10.1109/CVPR.2018.00340.

[3] S. Islam, A. Dash, A. Seum, A. H. Raj, T. Hossain, and F. M. Shah, "Exploring video captioning techniques: A comprehensive survey on deep learning methods," *SN Computer Science*, vol. 2, no. 2, Apr. 2021, doi: 10.1007/s42979-021-00487-x.

[4] O. Wiles, J. Carreira, I. Barr, A. Zisserman, and M. Malinowski, "Compressed vision for efficient video understanding," in *Computer Vision – ACCV 2022*, 2023, pp. 679–695, doi: 10.1007/978-3-031-26293-7_40.

[5] D. Alexandre and H.-M. Hang, "Learned video codec with enriched reconstruction for CLIC P-frame coding," *Computer Vision and Pattern Recognition*, Dec. 2020.

[6] Y. Tian *et al.*, "Self-conditioned probabilistic learning of video rescaling," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2021, pp. 4470–4479, doi: 10.1109/ICCV48922.2021.00445.

[7] M. Gorji, E. Hafezieh, and A. Tavakoli, "Advancing image deblurring performance with combined autoencoder and customized hidden layers," *Tuijin Jishu/Journal of Propulsion Technology*, vol. 44, no. 4, pp. 6462–6467, Oct. 2023, doi: 10.52783/tjjpt.v44.i4.2283.

[8] S. Yadav, C. Jain, and A. Chugh, "Evaluation of image deblurring techniques," *International Journal of Computer Applications*, vol. 139, no. 12, pp. 32–36, Apr. 2016, doi: 10.5120/ijca2016909492.

[9] K. Purohit, A. Shah, and A. N. Rajagopalan, "Bringing alive blurred moments," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2019, pp. 6823–6832, Apr. 2019, doi: 10.1109/CVPR.2019.00699.

[10] G. A. Farulla, M. Indaco, D. Rolfo, L. O. Russo, and P. Trotta, "Evaluation of image deblurring algorithms for real-time applications," in *2014 9th IEEE International Conference on Design & Technology of Integrated Systems in Nanoscale Era (DTIS)*, May 2014, pp. 1–6, doi: 10.1109/DTIS.2014.6850668.

[11] O. N. Gerek and Y. Altunbasak, "Key frame selection from MPEG video data," in *Visual Communications and Image Processing '97*, Jan. 1997, vol. 3024, pp. 920–925, doi: 10.1117/12.263304.

[12] M. Uhrina, J. Bienik, and M. Vaculik, "Subjective video quality assessment of H.265 compression standard for full HD resolution," *Advances in Electrical and Electronic Engineering*, vol. 13, no. 5, pp. 545–551, Dec. 2015, doi: 10.15598/aeee.v13i5.1503.

[13] M. M. Awad and N. N. Khamiss, "Low latency UHD adaptive video bitrate streaming based on HEVC encoder configurations and Http2 protocol," *Iraqi Journal of Science*, pp. 1836–1847, Apr. 2022, doi: 10.24996/ijs.2022.63.4.40.

[14] D. Watni and S. Chawla, "Enhancing embedding capacity of JPEG images in smartphones by selection of suitable cover image," in *ICDSMLA 2019*, vol. 601, 2020, pp. 211–220, doi: 10.1007/978-981-15-1420-3_22.

[15] Q. Meng, S. Zhao, Z. Huang, and F. Zhou, "MagFace: A universal representation for face recognition and quality assessment," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 2021, pp. 14220–14229, doi: 10.1109/CVPR46437.2021.01400.

[16] F. Kong and R. Henao, "Efficient classification of very large images with tiny objects," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2022, pp. 2374–2384, doi: 10.1109/CVPR52688.2022.00242.

[17] D. Smirnov and J. Solomon, "HodgeNet: learning spectral geometry on triangle meshes," *ACM Transactions on Graphics*, vol. 40, no. 4, pp. 1–11, Aug. 2021, doi: 10.1145/3450626.3459797.

[18] J. Liang, J. Cao, G. Sun, K. Zhang, L. Van Gool, and R. Timofte, "SwinIR: image restoration using swin transformer," in *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, Oct. 2021, pp. 1833–1844, doi: 10.1109/ICCVW54120.2021.00210.

[19] L. Tran, F. Liu, and X. Liu, "Towards high-fidelity nonlinear 3D face morphable model," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019, pp. 1126–1135, doi: 10.1109/CVPR.2019.00122.

[20] S. Niklaus, L. Mai, and F. Liu, "Video frame interpolation via adaptive separable convolution," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct. 2017, pp. 261–270, doi: 10.1109/ICCV.2017.37.

[21] K. C. K. Chan, X. Wang, K. Yu, C. Dong, and C. C. Loy, "BasicVSR: The search for essential components in video super-resolution and beyond," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2021, pp. 4945–4954, doi: 10.1109/CVPR46437.2021.00491.

[22] S. Nah, S. Son, and K. M. Lee, "Recurrent neural networks with intra-frame iterations for video deblurring," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019, pp. 8094–8103, doi: 10.1109/CVPR.2019.00829.

[23] V. Sharma, M. Gupta, A. Kumar, and D. Mishra, "Video processing using deep learning techniques: a systematic literature review," *IEEE Access*, vol. 9, pp. 139489–139507, 2021, doi: 10.1109/ACCESS.2021.3118541.

[24] H. Jiang, D. Sun, V. Jampani, M.-H. Yang, E. Learned-Miller, and J. Kautz, "Super SloMo: high quality estimation of multiple intermediate frames for video interpolation," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2018, pp. 9000–9008, doi: 10.1109/CVPR.2018.00938.

[25] J. Dong, K. Ota, and M. Dong, "Video frame interpolation: a comprehensive survey," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 19, no. 2s, pp. 1–31, Apr. 2023, doi: 10.1145/3556544.

[26] F. Reda *et al.*, "Unsupervised video interpolation using cycle consistency," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2019, pp. 892–900, doi: 10.1109/ICCV.2019.00098.

[27] H. Chen, M. Teng, B. Shi, Y. Wang, and T. Huang, "A residual learning approach to deblur and generate high frame rate video with an event camera," *IEEE Transactions on Multimedia*, vol. 25, pp. 5826–5839, 2023, doi: 10.1109/TMM.2022.3199556.

[28] Z. Huang, T. Zhang, W. Heng, B. Shi, and S. Zhou, "Real-time intermediate flow estimation for video frame interpolation," in *Computer Vision – ECCV 2022*, pp. 624–642, doi: 10.1007/978-3-031-19781-9_36.

BIOGRAPHIES OF AUTHORS

Redouane Lhiadi is Ph.D. student specializing in deep learning. He is a member of the Research Operations and Applied Statistics Team "ROSA" within the LaMAO Laboratory at the National School of Business and Management (ENCGO), University of Mohammed 1st in Oujda, Morocco. He can be contacted at email: lhiadi.redouane@gmail.com.

Dr. Abdessamad Jaddar is a professor and researcher at the National School of Business and Management (ENCGO) at the University of Mohammed 1st in Oujda, Morocco. He is a member of the Research Operations and Applied Statistics Team "ROSA" within the LaMAO Laboratory. He can be contacted at email: ajaddar@gmail.com.

Dr. Abdelali Kaaouachi is a full professor and director of a higher education institution, specializing in applied mathematics. His academic interests are diverse, focusing on decision-making tools such as probability, statistics, operational research, data analysis, and stochastic processes. He has conducted extensive research in rank-based statistical inference, developing new rank-based estimators for ARMA model parameters that outperform traditional estimators like the least squares estimator and the maximum likelihood estimator. His research also includes adaptive estimation, building upon the foundational work of Lucien Le Cam and Marc Hallin. He can be contacted at email: akaaouachi@hotmail.com.