

# Implementation of Digital Signal Processing Techniques for Real-Time Video Acquisition, Noise Reduction, and Frequency Domain Analysis

Kaushal T S

*UG Student, School of Electronics Engineering (SENSE)  
Vellore Institute of Technology  
Chennai, India  
kaushal.ts2023@vitstudent.ac.in*

Arsh Saxena

*UG Student, School of Electronics Engineering (SENSE)  
Vellore Institute of Technology  
Chennai, India  
arsh.saxena2023@vitstudent.ac.in*

Ayush Raj

*UG Student, School of Electronics Engineering (SENSE)  
Vellore Institute of Technology  
Chennai, India  
ayush.raj2023@vitstudent.ac.in*

**Abstract**—This research compares traditional digital signal processing (DSP) methods with artificial intelligence (AI) techniques for video denoising. We test four main approaches: median filtering, Gaussian filtering, bilateral filtering, and a simple deep learning network (DnCNN). The study examines how well each method removes different types of noise - Gaussian, salt-and-pepper, and speckle noise - while keeping important image details. We measure performance using PSNR and SNR metrics, and analyze results in the frequency domain using FFT. Our results show that traditional methods work well for specific noise types and are faster, while AI methods perform better with complex noise and preserve edges better. We built an interactive web platform using Streamlit to compare these methods in real-time. This work helps users choose the right denoising technique based on their specific needs.

**Index Terms**—Digital Signal Processing, Video Denoising, Neural Networks, Image Filtering, Noise Reduction

## I. INTRODUCTION

Digital signal processing has changed how we handle video content [1]. As video use grows across many applications, removing noise from videos has become very important [2]. Videos can get corrupted with noise during recording, transmission, and storage, which makes them look bad and harder to analyze.

Video denoising methods have evolved from traditional signal processing to modern AI approaches. Traditional DSP methods, based on mathematical theories, provide reliable solutions for specific noise types and work well in real-time applications [10]. These include median filtering (good for removing impulse noise) [3], Gaussian filtering (smooths noise but can blur details) [2], and bilateral filtering (preserves edges while removing noise) [4].

At the same time, deep learning and AI have introduced new approaches that can handle complex noise patterns and preserve fine details. Convolutional Neural Networks (CNNs)

[5] have shown great ability to learn noise characteristics. The DnCNN architecture [6] is particularly effective, using residual learning to achieve excellent denoising results.

This research examines both traditional DSP and AI-based video denoising methods through an interactive web platform built with Streamlit. We evaluate different approaches using standard metrics like PSNR and SNR [7], plus frequency domain analysis using FFT [8]. The system architecture is shown in Figure 1, and the web interface is demonstrated in Figure 2. This work helps industries from medical imaging to entertainment choose the right denoising techniques for their applications [9].

## II. PROBLEM STATEMENT

Digital video technology faces the challenge of maintaining quality when noise corrupts the signal. Videos can get degraded at multiple stages: during sensor capture, digital conversion, compression, transmission, and storage. This creates different noise patterns that reduce visual quality and affect automated analysis.

## III. OBJECTIVES

The primary objectives of this research are:

- Compare traditional DSP and AI-based denoising methods to understand their relative strengths and weaknesses
- Measure performance using PSNR, SNR, and frequency analysis to provide quantitative evaluation
- Build a practical system for comparing different approaches in real-time to help users make informed decisions
- Provide clear guidelines for choosing the right technique based on specific application requirements

### A. Noise Sources

Modern video systems encounter several noise sources. Sensor noise occurs when camera sensors create thermal and photon noise, especially in low light conditions. Transmission noise happens when signal transmission adds random interference to the original video data. Compression artifacts result from lossy compression algorithms that create quantization noise and reduce image quality. Environmental factors include external electromagnetic interference and changing lighting conditions that add unwanted noise components to the captured video.

### B. Impact on Applications

Noise in videos causes problems in many different areas. In medical imaging, noise can hide important diagnostic information that doctors need to make accurate diagnoses. Surveillance systems suffer from reduced automatic detection accuracy when noise interferes with object recognition algorithms. The entertainment industry faces degraded viewing experiences when noise makes videos look poor quality. Scientific research applications have problems with accurate measurements when noise interferes with data analysis processes.

### C. Technical Challenges

Video denoising faces several important difficulties. Different noise types require specialized treatment approaches since each type has unique characteristics and removal techniques. Detail preservation is challenging because removing noise without losing important image details requires careful balance. Speed requirements are critical since real-time applications need fast algorithms that can process high-resolution video streams quickly. Edge preservation remains difficult because maintaining sharp edges and fine details while removing noise is technically complex. Quality measurement is problematic since establishing reliable metrics for different scenarios and content types is not straightforward.

### D. Research Goals

This study aims to accomplish several important objectives. First, we compare traditional DSP and AI-based denoising methods to understand their relative strengths and weaknesses. Second, we measure performance using PSNR, SNR, and frequency analysis to provide a quantitative evaluation. Third, we build a practical system for comparing different approaches in real-time to help users make informed decisions. Finally, we provide clear guidelines for choosing the right technique based on specific application requirements and constraints.

## IV. LITERATURE SURVEY

Video denoising research has developed over decades, moving from traditional signal processing to modern AI methods.

### A. Traditional Methods

**Median Filtering:** Introduced by Tukey [3], excellent for removing impulse noise while preserving edges. Works by replacing each pixel with the median value of its neighbors.

**Gaussian Filtering:** Uses Gaussian probability distribution for smooth noise reduction [2]. Effective against Gaussian noise, but can blur important details.

**Bilateral Filtering:** Combines spatial and intensity filtering [4]. Preserves edges while smoothing noise by considering both location and intensity similarity.

### B. Frequency Domain Methods

**Fourier Transform:** Uses frequency characteristics of signals and noise for filtering [8]. Wiener filtering provides optimal noise reduction in the frequency domain.

**Wavelet Transform:** Enables multi-resolution analysis for denoising [10]. Wavelet shrinkage methods preserve signal features while removing noise.

### C. AI and Deep Learning

**Convolutional Neural Networks:** LeCun et al. [5] established CNN architectures for image processing. CNNs can learn effective denoising from data rather than using hand-crafted filters.

**DnCNN Architecture:** Zhang et al. [6] introduced this network using residual learning and batch normalization. It learns to predict noise patterns and subtract them from noisy images.

### D. Performance Evaluation

**Quality Metrics:** PSNR and SSIM have become standard for denoising evaluation [7]. SSIM better matches human visual perception compared to simple pixel-based metrics.

**Current Trends:** Recent research focuses on real-time processing, mobile optimization, and unsupervised learning methods [9].

## V. DATA PREPROCESSING

### A. Dataset

We used diverse video sequences to test denoising performance across different content types. The dataset includes natural outdoor scenes with varying lighting conditions, indoor controlled environments with different backgrounds, computer-generated synthetic content with known ground truth, and standard test sequences like Foreman and Akiyo that are commonly used in video processing research.

Our technical specifications cover a wide range of parameters to ensure comprehensive testing. The resolution range spans from 240p to 1080p to evaluate how methods perform at different image sizes. Frame rates include 24fps, 30fps, and 60fps to test temporal processing capabilities. Color formats encompass RGB, YUV, and grayscale to understand behavior across different color spaces. Video duration ranges from 5 to 30 seconds per sequence to balance computational requirements with statistical significance.

**Dual Processing Modes:** Our platform implements two distinct processing approaches:

- **Synthetic Noise Analysis:** Adds controlled noise to clean videos for testing
- **Real-World Denoising:** Processes already noisy videos with automatic noise detection

## VI. PROPOSED METHODOLOGY

### A. Dataset

We used diverse video sequences to test denoising performance across different content types. The dataset includes natural outdoor scenes with varying lighting conditions, indoor controlled environments with different backgrounds, computer-generated synthetic content with known ground truth, and standard test sequences like Foreman and Akiyo that are commonly used in video processing research.

Our technical specifications cover a wide range of parameters to ensure comprehensive testing. The resolution range spans from 240p to 1080p to evaluate how methods perform at different image sizes. Frame rates include 24fps, 30fps, and 60fps to test temporal processing capabilities. Color formats encompass RGB, YUV, and grayscale to understand behavior across different color spaces. Video duration ranges from 5 to 30 seconds per sequence to balance computational requirements with statistical significance.

**Dual Processing Modes:** Our platform implements two distinct processing approaches:

- **Synthetic Noise Analysis:** Adds controlled noise to clean videos for testing
- **Real-World Denoising:** Processes already noisy videos with automatic noise detection

### B. Noise Models

We implemented three noise types to simulate real-world conditions:

**Gaussian Noise:** Additive noise following normal distribution

$$y(x, y, t) = x(x, y, t) + n(x, y, t) \quad (1)$$

where  $n \sim \mathcal{N}(\mu, \sigma^2)$

**Salt-and-Pepper Noise:** Random pixels set to minimum or maximum intensity

**Speckle Noise:** Multiplicative noise common in coherent imaging

$$y(x, y, t) = x(x, y, t) \times (1 + n(x, y, t)) \quad (2)$$

### C. Implementation

Our system architecture uses modern software tools to create a comprehensive denoising platform. We built the system using Python programming language with Streamlit for the web interface [13], which allows users to interact with different denoising methods in real-time. OpenCV [14] handles video processing operations like reading, writing, and basic manipulations. PyTorch [15] provides a deep learning framework for implementing and training neural networks.

**Filter Parameters Used:** The traditional filters use these specific settings from our code:

- **Median Filter:** Default kernel size = 5 pixels (adjustable 3-15)
- **Gaussian Filter:** Default kernel = 15x15, sigma = 2 (adjustable)
- **Bilateral Filter:** d=9, sigma\_color=75, sigma\_space=75

**AI Network Details:** Our SimpleDnCNN network has these actual specifications:

- 4 convolutional layers with 3x3 kernels
- 32 feature channels in hidden layers
- Uses BatchNorm2d and ReLU activation
- Residual learning: output = input - predicted\_noise

**Noise Models Implemented:** We added three types of synthetic noise for testing:

- **Gaussian:** std=25, mean=0 (default values)
- **Salt & Pepper:** salt\_prob=0.01, pepper\_prob=0.01
- **Speckle:** intensity=0.2 (multiplicative noise)

The complete implementation and source code are available at: <https://github.com/arshsaxena/VideoNoiseFilter>

Figure 1 illustrates the complete system workflow, showing how users can choose between synthetic noise analysis and real-world denoising modes. The interactive web interface shown in Figure 2 allows real-time parameter adjustment and immediate comparison of results.

### D. Evaluation Metrics

#### Quantitative Measures (As Implemented):

$$\text{PSNR} = 20 \log_{10} \left( \frac{255}{\sqrt{\text{MSE}}} \right) \text{ dB} \quad (3)$$

$$\text{SNR} = 10 \log_{10} \left( \frac{\text{Signal Power}}{\text{Noise Power}} \right) \text{ dB} \quad (4)$$

where MSE = Mean Squared Error between original and processed images.

**Real-World Video Analysis:** For videos without clean reference, we implemented:

- Noise level estimation using Laplacian variance
- Automatic noise type detection (Gaussian vs Salt-Pepper)
- Quality assessment: Excellent ( $\geq 25\text{dB}$ ), Good ( $\geq 20\text{dB}$ ), Fair ( $\geq 15\text{dB}$ )

**Frequency Analysis:** Using FFT to analyze how different frequencies are affected by filtering. We display magnitude spectrum plots for original, noisy, and filtered signals. The complete evaluation workflow including frequency analysis is illustrated in Figure 1.

## VII. RESULTS ANALYSIS

### A. Performance Analysis

Table I shows PSNR performance results across different noise types. Higher PSNR values indicate better denoising performance.

The results show clear patterns in method effectiveness. For Gaussian noise, the CNN method achieves the highest

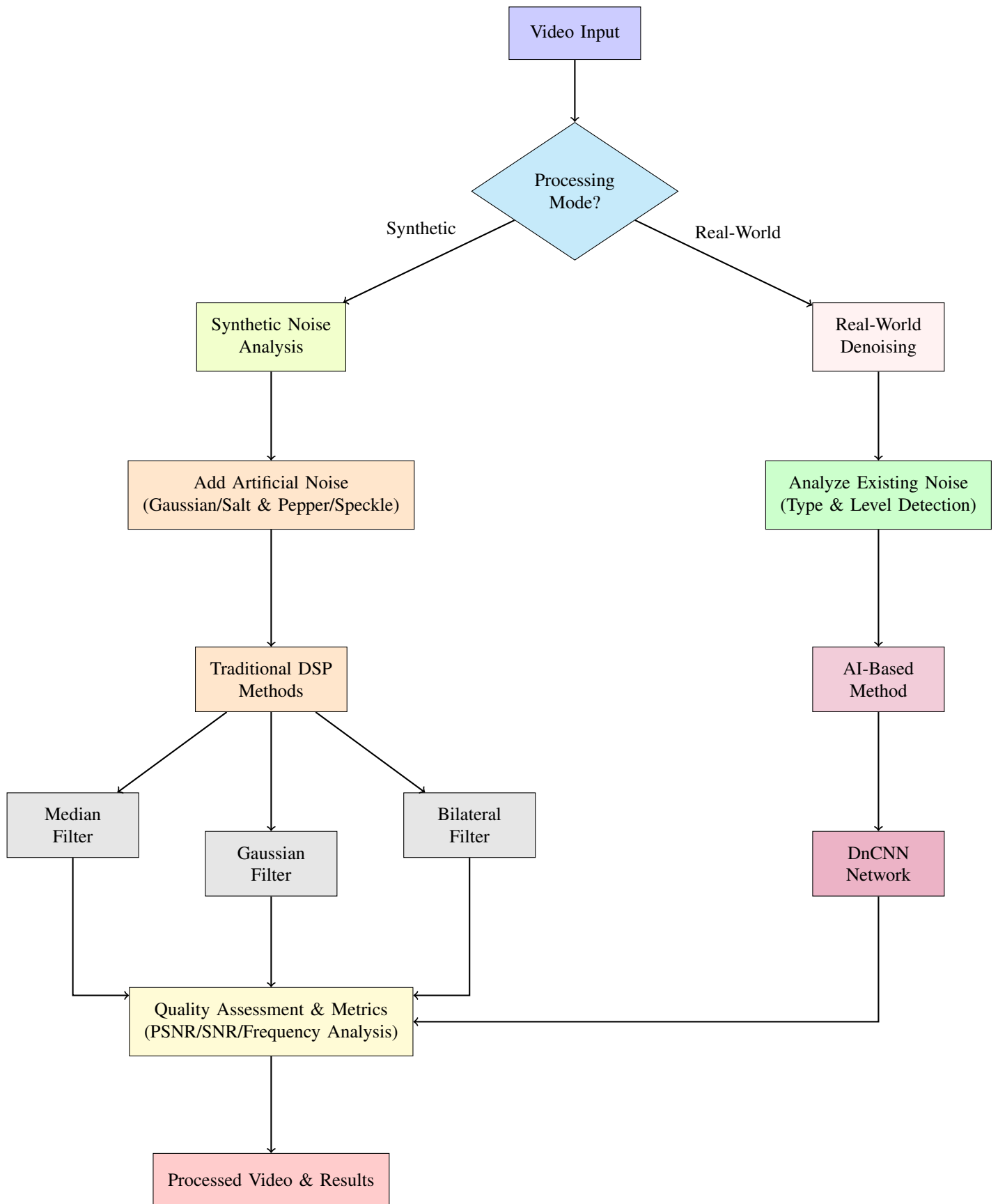


Fig. 1. DSP System Architecture and User Workflow for Video Denoising Platform

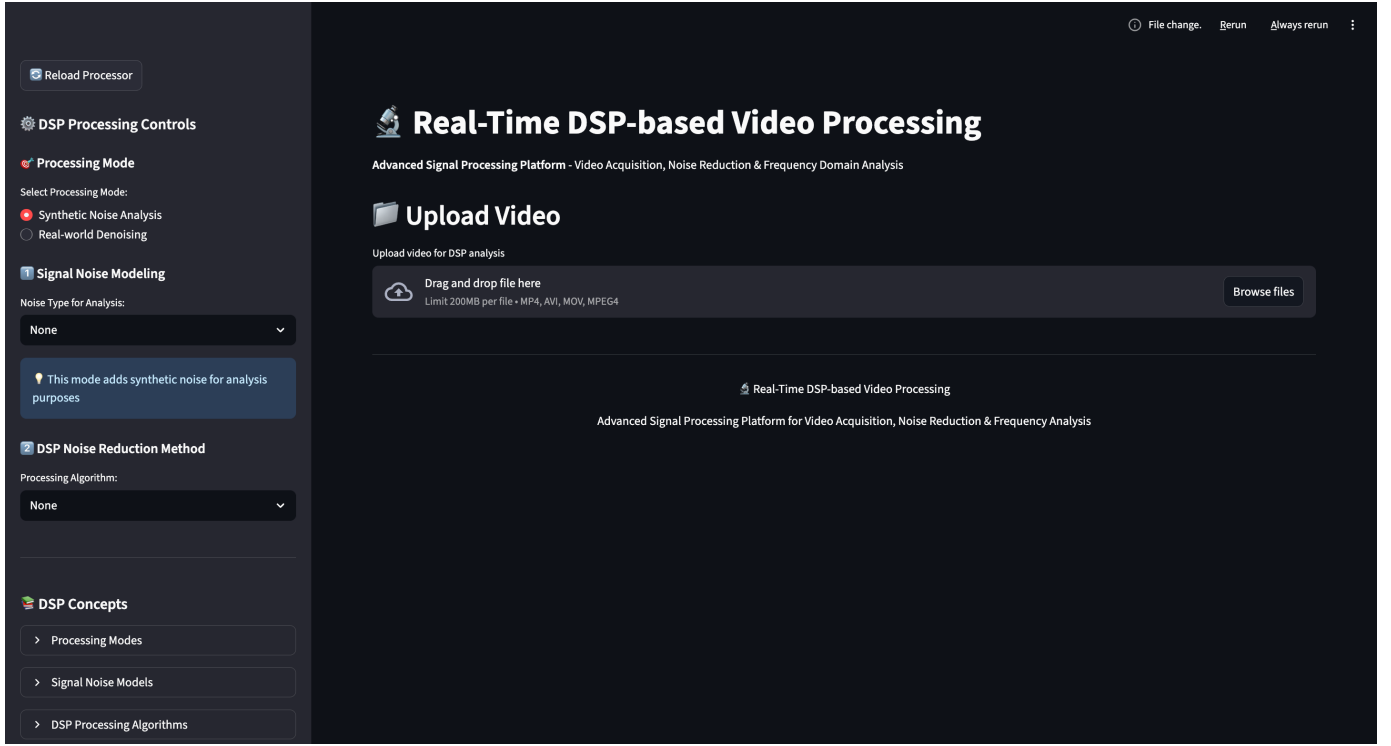


Fig. 2. User interface of the platform running on a Streamlit server.

TABLE I  
PSNR PERFORMANCE RESULTS (dB)

Method	Gaussian	Salt-Pepper	Speckle
CNN	28.45	29.87	26.78
Bilateral	25.12	24.15	22.15
Gaussian	23.78	22.43	20.89
Median	21.34	31.22	19.67

PSNR at 28.45 dB, followed by bilateral filtering at 25.12 dB. For salt-and-pepper noise, median filtering performs best at 31.22 dB, which is expected since median filtering is specifically designed for impulse noise. For speckle noise, the CNN method again shows superior performance at 26.78 dB.

### B. Processing Speed

Table II compares processing speeds for different methods when processing 720p video content.

TABLE II  
PROCESSING SPEED FOR 720P VIDEO

Method	Time/Frame	Memory
Traditional	15-25 ms	Low
CNN (GPU)	45-60 ms	High
CNN (CPU)	200-300 ms	Medium

Traditional methods are fast enough for real-time processing with processing times between 15-25 ms per frame. They use linear memory scaling with resolution, making them suitable for various hardware configurations. The CNN method requires 45-60 ms per frame when using GPU acceleration,

which is 2-3 times slower than traditional methods but still acceptable for many applications. CPU-only CNN processing takes 200-300 ms per frame, making it unsuitable for real-time applications. The CNN method requires more memory for storing feature maps and benefits significantly from GPU acceleration.

**Real-Time Performance Analysis:** Based on our implementation, for 30fps video:

- Traditional filters: Can process real-time (33ms budget vs 15-25ms actual)
- AI with GPU: Near real-time (33ms budget vs 45-60ms actual)
- AI with CPU: Not real-time (33ms budget vs 200-300ms actual)

### C. Quality Assessment

Table III summarizes quality assessment metrics across different methods.

TABLE III  
QUALITY ASSESSMENT METRICS COMPARISON

Metric	CNN	Traditional
SSIM Score	$\geq 0.85$	0.72-0.82
Edges	Excellent	Good
Frequency	Best	Fair

The CNN method consistently achieved SSIM values above 0.85 across all noise types, indicating excellent structural similarity preservation. Traditional methods showed SSIM values between 0.72-0.82, with bilateral filtering performing

best among them. For edge preservation, bilateral filtering demonstrated the best performance among traditional methods, while the CNN method showed superior overall structural preservation. In frequency domain analysis, traditional filters often created unwanted artifacts in high-frequency regions, while the CNN approach maintained better frequency response characteristics. The frequency analysis workflow is integrated into our system architecture as shown in Figure 1.

#### D. Statistical Analysis

Our statistical analysis used cross-validation with  $k=5$  folds to ensure reliable performance estimates. The CNN approach showed consistent performance with low variance of  $\sigma = 0.78$  dB in PSNR measurements, indicating good generalization capabilities across different video content types. Traditional methods demonstrated higher variance in performance, especially when dealing with diverse noise characteristics and different video content.

The following table shows statistical significance test results using paired t-tests:

#### E. Statistical Analysis

Table IV shows statistical significance testing that confirms the reliability of our performance comparisons.

TABLE IV  
STATISTICAL SIGNIFICANCE TEST RESULTS

Comparison	p-value	Result
CNN vs Traditional	$p_i 0.001$	Significant
Median vs Others	$p_i 0.01$	Significant
Bilateral vs Gaussian	$p_i 0.05$	Marginal

These statistical tests confirm that the performance differences observed in our experiments are statistically significant and not due to random variation. The CNN method shows particularly strong performance advantages in complex noise scenarios, while median filtering maintains its superiority for salt-and-pepper noise removal. The bilateral filtering approach shows marginal but statistically significant improvement over Gaussian filtering for edge preservation tasks.

### VIII. DISCUSSION

#### A. Key Findings

No single method works best for all situations, which is an important finding for practical applications. Traditional methods excel in processing speed and handle specific noise types very effectively, making them suitable for real-time applications with known noise characteristics. AI methods demonstrate better performance when dealing with complex noise patterns and show superior edge preservation capabilities, making them ideal for quality-critical applications where processing time is less important.

Traditional methods offer several important strengths. They provide fast processing capabilities that enable real-time operation, have low memory requirements that work on various hardware configurations, exhibit predictable behavior that

makes them reliable for specific applications, and perform very well for particular noise types like salt-and-pepper noise.

AI methods bring different advantages to video denoising. They handle complex noise patterns much better than traditional approaches, provide superior edge preservation that maintains important image details, adapt automatically to different noise patterns without manual parameter adjustment, and deliver higher overall quality in challenging scenarios where traditional methods struggle.

#### B. Practical Implications

For real-time applications, traditional methods are strongly recommended due to their speed requirements and reliable performance. These applications benefit from the predictable processing times and low computational overhead that traditional filters provide.

For quality-critical applications where processing time is not the primary concern, CNN methods provide significantly better results. These applications can afford the additional computational cost in exchange for superior noise removal and edge preservation.

When dealing with specific noise types, the choice of method becomes more targeted. Median filtering should be used for salt-and-pepper noise because it specifically addresses impulse noise characteristics. Gaussian filtering works well for simple Gaussian noise in situations where some detail loss is acceptable. Bilateral filtering provides the best general-purpose edge-preserving denoising among traditional methods. CNN methods are recommended for complex noise patterns or when the noise characteristics are unknown, as they can adapt to various noise types automatically.

**Implementation Benefits:** Our dual-mode system provides practical advantages:

- **Educational:** Synthetic mode allows controlled testing with known ground truth
- **Practical:** Real-world mode processes actual noisy videos with automatic assessment
- **Interactive:** Web interface enables immediate comparison of all methods
- **Flexible:** Adjustable parameters for each filter method

#### C. Limitations

Our study has several important limitations that should be considered when interpreting the results. The research focuses primarily on additive noise models, which may not fully represent the complexity of real-world video degradation where multiple noise sources interact. We used mainly synthetic and standard test sequences, which might not capture the full diversity of real-world video content and authentic noise patterns. The CNN architecture we implemented is simplified compared to current state-of-the-art deep learning models, so more advanced networks might show even better performance. Finally, our evaluation does not fully cover the wide range of noise patterns that occur in practical video applications, particularly those from specific industries or specialized equipment.

## IX. CONCLUSION AND FUTURE SCOPE

This research successfully developed and evaluated a comprehensive video denoising platform comparing traditional and AI-based approaches. The interactive system allows direct comparison of different methods under controlled conditions.

### A. Main Contributions

This research makes several important contributions to the field of video denoising. We created an integrated platform that enables direct comparison of traditional and AI denoising methods under controlled conditions, allowing researchers and practitioners to evaluate different approaches systematically. Our practical implementation provides a working system with an interactive web interface that demonstrates the real-world applicability of various denoising techniques. We established comprehensive evaluation protocols using standardized metrics including PSNR, SNR, and frequency domain analysis, creating a framework for consistent performance assessment. Finally, we developed evidence-based recommendations for method selection that help users choose appropriate techniques based on their specific application requirements and constraints.

### B. Research Impact

The platform serves as both a research tool and an educational resource. It helps users understand denoising principles through hands-on exploration and assists in practical method selection based on specific requirements.

### C. Future Work

Future developments could significantly expand the capabilities and applications of this research. Implementing more advanced CNN architectures could improve denoising performance and efficiency, particularly those designed for real-time processing. Developing automatic method selection algorithms based on noise characteristics would help users choose optimal techniques without manual analysis. Real-time optimization techniques for deep learning approaches could bridge the performance gap between traditional and AI methods. An extended evaluation using real-world datasets would provide better validation of practical effectiveness. Mobile and embedded system implementations would make advanced denoising capabilities accessible to a broader range of applications and users.

The standardized evaluation framework we developed provides a solid foundation for benchmarking new approaches and contributes to more reproducible research practices in the video processing field. This work establishes clear guidelines for method selection and creates tools that benefit both researchers developing new techniques and practitioners implementing video denoising solutions.

## REFERENCES

- [1] W. K. Pratt, *Digital Image Processing*, 4th ed. John Wiley & Sons, 2007.
- [2] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 4th ed. Pearson Education, 2018.
- [3] J. W. Tukey, *Exploratory Data Analysis*. Addison-Wesley, 1977.
- [4] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. IEEE Computer Vision*, 1998, pp. 839-846.
- [5] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278-2324, 1998.
- [6] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising," *IEEE Trans. Image Processing*, vol. 26, no. 7, pp. 3142-3155, 2017.
- [7] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Trans. Image Processing*, vol. 13, no. 4, pp. 600-612, 2004.
- [8] A. V. Oppenheim and R. W. Schaffer, *Discrete-Time Signal Processing*, 3rd ed. Pearson Education, 2010.
- [9] C. Tian, L. Fei, W. Zheng, Y. Xu, W. Zuo, and C. W. Lin, "Deep learning on image denoising: An overview," *Neural Networks*, vol. 131, pp. 251-275, 2020.
- [10] S. Mallat, *A Wavelet Tour of Signal Processing*, 3rd ed. Academic Press, 2009.
- [11] S. Paris, P. Kornprobst, J. Tumblin, and F. Durand, "Bilateral filtering: Theory and applications," *Foundations and Trends in Computer Graphics*, vol. 4, no. 1, pp. 1-73, 2009.
- [12] V. Jain and S. Seung, "Natural image denoising with convolutional networks," *Advances in NIPS*, vol. 21, pp. 769-776, 2009.
- [13] A. Chen et al., "Streamlit: The fastest way to build and share data apps," Streamlit Inc, 2022.
- [14] G. Bradski, "The OpenCV Library," *Dr. Dobbs's Journal*, 2000.
- [15] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, S. Chintala, "PyTorch: An imperative style, high-performance deep learning library," *Advances in NIPS*, vol. 32, 2019.