ReTiDe: Real-Time Denoising for Energy-Efficient Motion Picture Processing with FPGAs

Anonymous Author(s)

(a) Ground truth (b) Noisy (c) ReTiDe (F32) (d) ReTiDe (QAT)

Figure 1: In the NTIRE25 dataset, when the noise intensity is 50, the ground truth, noisy image and denoising results from the FP32 and the quantised ReTiDe.

Abstract

Denoising is a core operation in modern video pipelines. In codecs, in-loop filters suppress sensor noise and quantisation artefacts to improve rate-distortion performance; in cinema post-production, denoisers are used for restoration, grain management, and plate clean-up. However, state-of-the-art deep denoisers are computationally intensive and, at scale, are typically deployed on GPUs, incurring high power and cost for real-time, high-resolution streams. This paper presents Real-Time Denoise (ReTiDe), a hardware-accelerated denoising system that serves inference on data-centre FPGAs. A compact convolutional model is quantised (post-training quantisation plus quantisation-aware fine-tuning) to INT8 and compiled for AMD DPU-based FPGAs via Vitis-AI. A client-server integration offloads computation from the host CPU/GPU to a networked FPGA service, while remaining callable from existing workflows, e.g., NUKE, without disrupting artist tooling. On representative benchmarks, ReTiDe delivers 37.71× throughput (GOPS) and 4.42× higher energy efficiency than prior FPGA denoising accelerators, with negligible degradation in Peak Signal-to-Noise Ratio (PSNR)/Structural Similarity Index (SSIM). These results indicate that specialised accelerators can provide practical, scalable denoising for both encoding pipelines and post-production, reducing energy per frame without sacrificing quality or workflow compatibility. The code will be made publicly available upon acceptance.

CCS Concepts

• **Computing methodologies** → **Image processing**; *Neural networks*; • **Hardware** → *Hardware accelerators*; Power and energy.

Keywords

Image, Denoising, Deep Learning, FPGAs

CVMP 2025, London, UK
2025. ACM ISBN 978-1-4503-XXXX-X/2025/12
https://doi.org/XXXXXXX.XXXXXXX

ACM Reference Format:
Anonymous Author(s). 2025. ReTiDe: Real-Time Denoising for Energy-Efficient Motion Picture Processing with FPGAs. In *Proceedings of the 22nd ACM SIGGRAPH European Conference on Visual Media Production (CVMP 2025).* ACM, New York, NY, USA, 10 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Image denoising is a classic ill-posed problem with applications in video compression [Brenig and Timofte 2025], cinematic post-production [Bled and Pitié 2024], camera and smartphone imaging pipelines [Wang et al. 2024], and medical imaging [Demir et al. 2025]. In video compression, denoising reduces frame entropy, enabling more efficient encoding and lower bandwidth usage. In cinematic post-production, it is essential to clean raw footage before editing and grading. In smartphone imaging, denoising mitigates the higher noise levels inherent to small sensors. In medical imaging, it improves diagnostic clarity by suppressing acquisition noise.

In recent years, image denoising has shifted from classic methods [Bled and Pitie 2022] based on analytical priors, such as regularisation [Combettes and Pesquet 2004; Malfait and Roose 1997; Mallat 1989], wavelet-based denoisers [Combettes and Pesquet 2004; Malfait and Roose 1997; Mallat 1989], and nonlocal collaborative filters [Buades et al. 2005; Coupé et al. 2008; Gilboa and Osher 2009; Mahmoudi and Sapiro 2005; Wang et al. 2006], including the widely used BM3D algorithm [Dabov et al. 2007, 2009], to data-driven deep learning approaches [Guo et al. 2019; Gurrola-Ramos et al. 2021; Heinrich et al. 2018; Liu et al. 2018; Qin et al. 2020; Wang et al. 2020; Zamir et al. 2021; Zhang et al. 2017a, 2018] capable of restoring image details that are otherwise irretrievably lost. While these modern methods generally outperform traditional filters, they often come with substantially higher computational costs, with state-of-the-art transformer architectures [Dosovitskiy et al. 2020; Liang et al. 2021; Liu et al. 2021; Vaswani et al. 2017] capable of employing hundreds of millions of trainable parameters.

Although research on general-purpose FP32 GPU-accelerated models continues to advance, studies on energy-efficient hardware-accelerated quantised denoising models in this field are limited. Studies have shown that, with appropriate modifications, quantised neural networks can achieve comparable performance with significantly reduced overhead [Han et al. 2016]. FPGAs are an ideal offloading platform for many compute-intensive tasks, including video denoising, due to their latency and power efficiency. Existing efforts of FPGA-accelerated denoising have mainly focused on optimising classical bilateral filters [Dabhade et al. 2017; Gabiger-Rose et al. 2013; Spagnolo et al. 2024; Wen et al. 2024], whereas research on accelerating quantised deep learning–based denoising models remains largely underexplored. Recently, studies have demonstrated that deploying deep neural networks such as DnCNN on FPGAs can outperform CPU and GPU implementations in terms of throughput and energy efficiency [Kang et al. 2024; Tu et al. 2024]. These approaches rely on specialised quantised convolution IPs based on algorithms such as Winograd fast convolution, which reduce the number of multiplications through matrix transformations and are particularly effective for specific kernel sizes (especially 3×3). These optimisations make them more efficient in both computation and energy consumption. However, such fixed architectures are less suited to advanced models and lack highly parallel pipelined scheduling. Their intermediate results require high-bit storage, which constrains both throughput and energy efficiency.

While post-production workflows now offer FP32 GPU accelerated models, such as The Foundry's catalogue (CATTERY) of deep-learning models [The Foundry Visionmongers Limited 2025], the advantages of quantised neural network models on FPGAs (in terms of throughput and energy efficiency), particularly for intensive media stream processing, have never been incorporated. To address this, we propose an end-to-end video denoising framework that enables professional users to offload denoising tasks to cloud-based FPGAs via a simple interface, achieving technological decoupling and high-efficiency, low-power video denoising.

The main contributions of this paper are as follows:

- We use Quantisation-Aware Training (QAT) to convert FP32 models to INT8 while maintaining image quality, delivering substantial gains in throughput and energy efficiency with negligible PSNR loss.
- We create a NUKE plugin that invokes a networked FPGA denoising service, offloading denoising work, enabling real-time use without disrupting artist workflow.
- The quantised ReTiDe-Net achieves denoising quality comparable to popular existing FP32 models.
- To the best of our knowledge, ReTiDE is the first open-source, colour, blind, hardware-accelerated denoiser.

The remainder of this paper is organised as follows. Section 2 reviews the background and related work, including image noise characteristics, recent advances in AI-based methods, and prior research on FPGA-based denoising. Section 3 presents our proposed Client-Server denoiser integration framework, detailing the quantised denoising model and deployment strategies. Section 4 discusses the experimental results, comparing the performance of our model with state-of-the-art methods on both colour and grayscale image denoising tasks. Finally, Section 5 concludes the paper.

2 Background and Related Works

2.1 Noise in Digital Photography

Although digital camera technology continues to advance, the quantum nature of light imposes a fundamental noise floor in every image. Due to quantum uncertainty, the arrival of photons at discrete photowells follows Poisson statistics, leading to unavoidable fluctuations in the measured photon counts from one photowell to another. This phenomenon, known as photon noise [Beenakker and Schönenberger 2003; Schottky 1918, 2018] (or shot noise), originates outside the sensor's silicon and defines the minimum achievable noise level in an image. Subsequent amplification of the captured signal within the camera's image signal processor (ISP) introduces additional noise sources, including dark current noise [Yang and Gamal 1999], fixed-pattern noise [Joseph and Collins 2001], and read noise [Liu and Gamal 2001].

2.2 Popular Denoising Algorithms

Since its introduction in the late 2000s, BM3D [Dabov et al. 2007] has remained a leading classical image denoiser. It performs collaborative filtering of non-local patches in two stages: first, grouping similar patches via block matching, weighting them by similarity, and applying hard thresholding in a transform domain to produce a pilot estimate; second, refining the estimate with a collaborative Wiener (or later, wavelet) filter. Variants of these algorithmic approaches remain in use, for example, Neat Video's [GmbH 2024] wavelet-based filter and Wiener-based denoisers in The Foundry's NUKE [Ltd. 2023] and the AV1 [for Open Media nd] in-loop filter.

In 2017, the deep learning model DnCNN [Zhang et al. 2017a] demonstrated clear gains over classical methods by employing an end-to-end 17-layer (3×3 CONV+BN+ReLU) convolutional network without downsampling. The network predicts the residual image, which is subtracted from the input to produce the denoised output. Its compact size (557k parameters) has contributed to its enduring popularity. Subsequent models built on this foundation, such as IRCNN [Zhang et al. 2017b] with dilated convolutions and FFDNet [Zhang et al. 2018], which incorporates user-provided noise levels and subsampled inputs.

The encoder–decoder architecture of U-Net [Ronneberger et al. 2015] enabled much larger denoising networks, improving quality through multi-scale feature analysis and fusion via skip connections. CBDNet [Guo et al. 2019] (4M parameters) extends U-Net for blind denoising by incorporating a noise estimation subnetwork alongside the noisy input. MWCNN [Liu et al. 2018] and MWRDCNN integrate wavelet transforms into U-Nets, decomposing features into high- and low-frequency components for more efficient processing, and remain among the most competitive non-transformer U-Net variants. Other widely used U-Net denoisers include MPRNet [Zamir et al. 2021] (20M parameters) and U2Net [Qin et al. 2020] (44M parameters), reflecting the architecture's adaptability across diverse denoising tasks.

Most recently, transformer networks [Dosovitskiy et al. 2020; Khan et al. 2022; Vaswani et al. 2017] have surpassed the performance of CNNs by introducing multi-headed attention layers with global receptive fields, enabling them to capture long-range dependencies beyond local convolutions. The Swin Transformer [Liu et al. 2021] established a general-purpose backbone by using hierarchical

attention layers that progressively downsample features and employ shifted windows to achieve global context, making end-to-end image tokenisation and reconstruction computationally tractable. This architecture has since inspired several denoising networks, including SwinIR [Liang et al. 2021], Uformer [Wang et al. 2022], and Restormer [Zamir et al. 2022].

2.3 Hardware Accelerated Video Denoising

Conventional denoisers are typically implemented on CPUs or GPUs. However, their low throughput and limited energy efficiency often become performance bottlenecks in practical applications. Recently, a growing body of research has explored offloading both classical [Dabhade et al. 2017; Gabiger-Rose et al. 2013; Spagnolo et al. 2024; Wen et al. 2024] and deep learning-based image denoising algorithms [Kang et al. 2024; Tu et al. 2024] onto FPGAs to improve performance. Compared to other platforms, FPGAs offer superior throughput, energy efficiency, and reconfigurability, making them an excellent offloading target for such tasks.

Research on optimising hardware-accelerated image denoising algorithms is limited, with even recent work [Spagnolo et al. 2023, 2024; Wen et al. 2024; Xie et al. 2024; Yao et al. 2022] remaining limited to traditional bilateral filter implementations. While these examples do not compete with state-of-the-art GPU models in terms of image quality, their simple implementations offer fast, real-time results. Several studies have proposed accelerating the bilateral filtering process by simplifying the computation, approximating the filter kernels, and increasing parallelism, which significantly enhances filtering efficiency. In [Dabhade et al. 2017], a constant-time bilateral filtering algorithm using Gaussian polynomial approximation for the spatial kernel was deployed on an FPGA. This approach enables the use of larger kernels without additional resource overhead. Gabiger et al. [Gabiger-Rose et al. 2013] achieved pipelined denoising on an FPGA through pixel grouping and clock-level acceleration. Wen et al. [Wen et al. 2024] reduced computational complexity via approximate computing and improved throughput for high-resolution image processing using a data prefetching strategy. Similarly, Fanny et al. [Spagnolo et al. 2024] improved energy efficiency using approximation techniques, while preserving real-time filtering performance and high visual precision.

Storing filter weights in lookup tables (LUTs) allows for precomputation on hardware, reducing computational cost. Spagnolo et al. [Spagnolo et al. 2023] approximated the coefficients of both kernels using piecewise functions and encoded them as 7-bit unsigned integers, storing them in reduced-size LUTs. For a 55 kernel, their implementation achieved a maximum operating frequency of 244 MHz and a throughput of 926.8 frames per second. In [Yao et al. 2022], Yao et al. proposed a low-cost bilateral filter hardware architecture incorporating a LUT-based divider and a parallelised design, capable of processing 8-megapixel video at 30 frames per second. The implementation of the Gaussian-Adaptive Bilateral Filter (GABF) on FPGA further demonstrates that kernel approximation and pipelining can effectively accelerate the denoising process while maintaining real-time performance [Xie et al. 2024].

Recent advances in deep learning for image denoising have shown that its performance has gradually surpassed traditional methods, especially in detail processing, prompting researchers to explore the use of Quantised Neural Networks (QNNs) in denoising accelerators. In recent studies, Tu et al. [Tu et al. 2024] achieved 5.39× and 15.23× higher energy efficiency compared to GPU and CPU implementations, respectively, by deploying TNet on a ZYNQ FPGA (MZU03A-EG).Similarly, Kang et al. [Kang et al. 2024] implemented a quantised DnCNN for denoising, achieving 1.9× and 26.2× energy efficiency improvements over GPU and CPU baselines, respectively. These studies highlight the potential of applying QNNs for denoising on FPGAs. However, their primary focus lies in inference efficiency and hardware performance, lacking a comprehensive denoising performance evaluation. TNet-mini provides only limited case studies of denoising on small-scale datasets, without presenting statistical evaluations of denoising performance, such as PSNR comparisons before and after denoising under different noise environments. L-DnCNN reports PSNR results only on the small-scale grayscale datasets BSD68 and SET12, and only for limited noise levels (15, 25, and 50), without evaluating noise reduction performance on colour images or assessing the method on higher-quality datasets. Furthermore, these Winograd-based convolutional [Kala et al. 2019] accelerators lack flexibility and have poor parallelism, leaving much room for improvement in hardware performance. Additionally, neither implementation has been open-sourced.

To address these limitations, we implement the ReTiDe denoiser, starting from the Cycle-GAN generator [Zhu et al. 2017], optimising the network for hardware and finally quantising and retraining for hardware. Leveraging highly parallel DPU acceleration, the proposed approach further exploits the real-time and energy-efficient characteristics. We purposely train blind models for both colour and grayscale, unlike many existing models, which require user input or the selection of a model trained within a certain noise band. The quantised denoisers are benchmarked under multiple noise levels for multiple datasets, demonstrating their generalisation capability. Finally, an end-to-end deployment interface integrated with a server-accelerator architecture, bridging the gap for professional image processing users in effectively harnessing the advantages of hardware acceleration.

3 Methodology

3.1 Lightweight ReTiDe-Net

QNNs significantly reduce model size and improve execution efficiency by converting parameters from FP32 to INT8 representations. Moreover, mature FP32 operator libraries on GPUs can be mapped to more efficient quantised implementations, where techniques such as operator substitution (e.g., replacing multiplication with shift operations) and operator fusion (e.g., convolution + batch normalisation + activation) transform them into hardware-friendly forms. This greatly enhances hardware efficiency and enables acceleration on fundamental hardware units such as DSPs. However, discrepancies between operators and the delayed implementation of quantised operators pose challenges for mapping advanced operations and model structure selection.

Considering this trade-off, we adopt the cGAN backbone [Zhu et al. 2017], which has been demonstrated as suitable for hardware conversion [Murphy et al. 2023], and adapt its U-Net generator while discarding the discriminator. The structure of the model is

shown in Figure 2. Our generator is a symmetric encoder–decoder with skip connections between matching stages. In contrast to the original eight-stage design, we employ six downsampling stages, each implemented with a single strided convolution (no bias), paired with a corresponding transposed convolution for upsampling. The innermost block contains both a convolution and a transposed convolution, forming the bottleneck. Skip connections are realised by concatenating encoder features with the decoder output at the same resolution. To improve hardware efficiency, all downsampling layers employ LeakyReLU activations, which prevent gradient vanishing and feature loss that could result from the absence of negative values during the downsampling process. Furthermore, using a fixed-slope ($\alpha = 0.1015625$) LeakyReLU enables accurate mapping onto hardware Look Up Table (LUT) operations, avoiding additional Block Random Access Memory (BRAM) and Digital Signal Processing block (DSP) resource consumption, thus ensuring both quantisation accuracy and high energy efficiency. During upsampling, more regularised feature distributions allow the use of ReLU activations, which enhances decoding efficiency. Normalisation, dropout, and residual connections are omitted to simplify the design and reduce hardware cost. With this design strategy, we train a grayscale and a colour model and convert both to hardware to compare their performance to existing models.

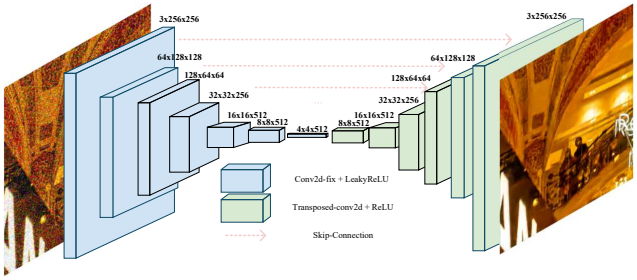


Figure 2: ReTiDe model structure.

3.2 Accelerator Quantisation and Deployment

The integration workflow of Vitis AI with NUKE is illustrated in Figure 3. The AMD Vitis AI toolchain provides an end-to-end workflow for deploying quantised neural networks, bridging the gap between machine learning frameworks and FPGA-based deployment. Starting from FP32 model descriptions written in popular frameworks such as TensorFlow and PyTorch, the toolchain performs model quantisation and operator conversion. The converted models can then be accelerated on the Deep Learning Processing Unit (DPU), which is specifically designed for convolutional and matrix-intensive workloads. By mapping computation-intensive kernels directly onto dedicated hardware engines, the toolchain not only reduces CPU overhead but also maximises parallelism and memory bandwidth utilisation. This hardware–software co-design approach significantly improves inference throughput while simultaneously reducing power consumption.

After training a 32-bit PyTorch FP32 model, Post-Training Quantisation (PTQ) is first applied to convert the original FP32 weights

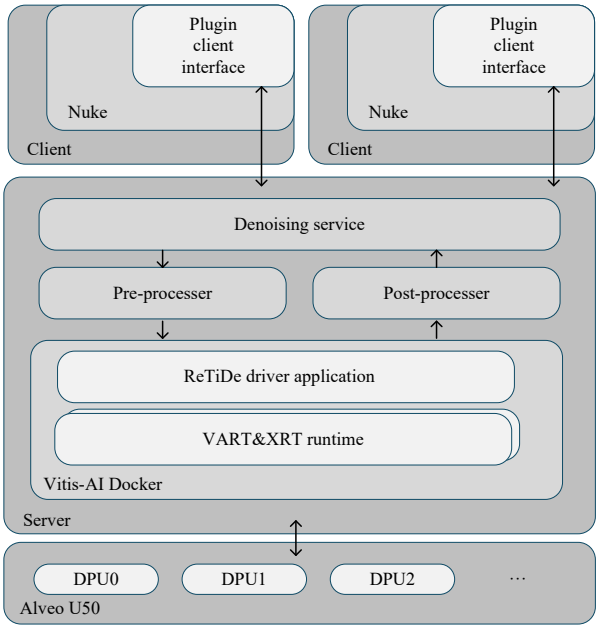


Figure 3: Diagram of the Vitis-NUKE integration.

and activations into an 8-bit fixed-point format. PTQ utilises representative data during the calibration phase to capture the distribution characteristics of activations, and then computes scaling factors and zero-point parameters to map continuous FP32 values into a finite integer range. This process substantially reduces model storage overhead and bandwidth requirements, while also allowing the model to better adapt to the hardware logic resources of the FPGA. However, since quantisation inevitably introduces rounding errors and numerical discretisation, model accuracy may be affected. Only calibration is insufficient to adapt the model to quantised inference.

QAT was introduced to mitigate the accuracy degradation caused by quantisation. In the computational graph, we inserted quantisation stubs and replaced some operators explicitly, resulting in a trainable pseudo-quantised model that emulates quantisation effects during training. During forward propagation, these nodes approximate computations for weights and activations, while in backwards propagation, gradients are still computed using FP32 parameters. Through subsequent fine-tune retraining, the pseudo-quantised model adapts to perturbations introduced by quantisation, thereby effectively restoring denoising performance. A small subset of data is utilised for several forward passes to provide statistical calibration when exporting the quantisation configuration. Finally, the model is quantised and exported for further compilation.

The quantised denoising U-Net model and its quantised weights are loaded into the Vitis AI Docker container for compilation, generating the DPU executable file (xmodel), which can be efficiently mapped onto DPU acceleration tasks on AMD Alveo FPGA accelerator cards. The Alveo platform integrates high-bandwidth memory, fast interfaces, and dedicated management engines, providing strong support for large-scale image processing and inference tasks

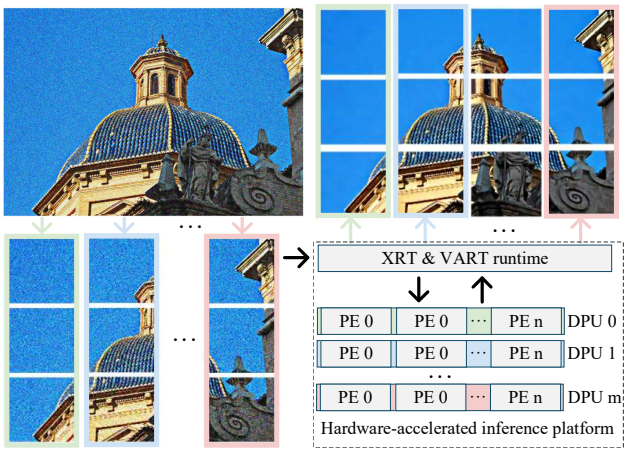


Figure 4: Pre-processing of large input images, parallel hardware-accelerated noise reduction and post-processing.

in data centre environments. To achieve seamless integration with practical rendering workflows, we developed customised runtime drivers that provide an interface between the DPU and the quantised model. This allows the NUKE rendering system to perform multi-threaded parallel invocations through the server-side processing engine, thereby significantly accelerating the inference process while maintaining high accuracy.

3.3 NUKE Software Interface Integration

The NUKE Machine Learning Plugin is a dedicated toolkit developed specifically for NUKE, enabling the incorporation of machine learning models into its node-based VFX software environment. To facilitate the integration of the noise reduction function, the noise reduction service interface is encapsulated as a type of NUKE plug-in. Additionally, by modifying the client prior to compilation, the message buffer is extended to accommodate 8K-level data streams. Distributed client hosts can transmit target data to designated servers for real-time rendering by invoking remote deep learning processing services.

The server side consists of a host machine equipped with an Alveo U50 server-level FPGA accelerator card. Figure 4 demonstrates a processing flow from the original noisy image ($\sigma = 50$) to the denoised image with segmentation and corresponding parallelisation. Upon receiving denoising requests initiated by remote or local hosts, the incoming image or video stream is first processed by a pre-processor, which segments and batches the media into standardised input formats suitable for the model. This also facilitates parallel processing across multiple threads and DPU units.

The Vitis AI runtime and associated drivers are encapsulated within a standalone Docker image, providing a stable runtime environment and reducing deployment costs. The denoising service invokes the model driver to batch-process the denoising sequences through the VART and XRT runtimes. These sequences are offloaded to the FPGA accelerator via the high-speed PCIe interface in a multithreaded fashion. Subsequently, tasks are distributed in parallel across multiple DPUs optimised for quantised convolution operations, where further parallelisation is achieved through multiple

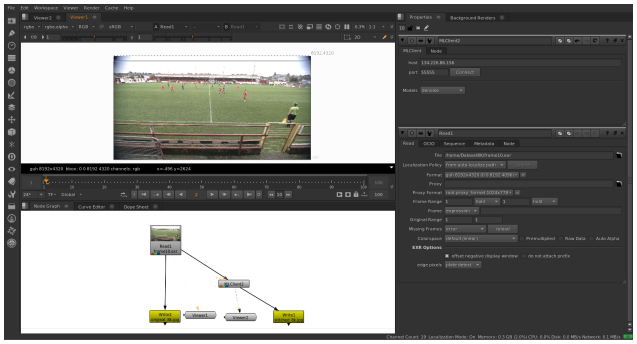


Figure 5: NUKE User Interface.

processing elements (PEs). This hierarchical and dedicated parallel processing architecture significantly enhances system throughput and energy efficiency. The post-DPU output is then passed back in reverse order, undergoing post-processing to restore the spatial and temporal sequence before being delivered back to the client.

The user interface of the client is shown in Figure 5, where an 8K image denoising example is used for illustration. The upper-left, lower-left, and upper-right sections of the interface respectively display the media to be processed, the state machine diagram of the processing workflow, and the connection configuration for the denoising model server. In the state machine diagram, Viewer1 and Viewer2 are used to preview the input and output of the denoising process, respectively. Additionally, the processed results are stubbed for further access or inspection. Through the denoising service, the complex hardware-accelerated denoising functionality is abstracted into a simple, callable function block that can be easily integrated and triggered within the automated state machine workflow.

4 Results and Discussion

4.1 Dataset and Experimental Setups

The DIV2K [Agustsson and Timofte 2017; Timofte et al. 2017] and LSDIR [Li et al. 2023] datasets are used for training. Together, they comprise over 85,000 images, although we found that a subset of 4,000 images was sufficient for training the denoiser. These datasets were selected for their high spatial resolution (2K and 4K) and their greater diversity compared to classic computer vision datasets, which typically contain far fewer images at lower resolutions and are often limited to film photography content.

Training is performed on randomly cropped 256×256 patches with a batch size of 64 for 10,000 epochs. For each batch, patches are sampled from the full-resolution images and augmented with random horizontal and vertical flips and random rotations. Optimisation is carried out using AdamW with an initial learning rate of $\eta = 10^{-4}$ and weight decay of $\lambda = 10^{-2}$, with a cosine annealing scheduler that decays the learning rate to zero and restarts every 5,000 batches. Model training is conducted in FP32 on an NVIDIA A5000 GPU, using Python 3.13, PyTorch 2.1.1, and CUDA 12.2.

The grayscale versions of datasets BSD68 [Martin et al. 2001], Urban100 [Huang et al. 2015] and Set12 [Dabov et al. 2007] are used to evaluate the grayscale model and to fairly compare against the existing models [Kang et al. 2024; Tu et al. 2024]. Colour benchmarks

are carried out on BSD100 [Martin et al. 2001]. In the QAT process, we retrained the model for 30 epochs in a quantise-aware manner with a learning rate of 10^{-8} to fine-tune and recover the PSNR. Lastly, the Alveo-U50 FPGA was selected as the targeted platform.

To evaluate the performance of our denoising model against existing FPGA-accelerated denoising approaches, we conducted experiments from the following perspectives: (1) Comparison of denoising performance in grayscale image denoising with both state-of-the-art FP32 models and FPGA-based deep-learning-accelerated quantised denoising models. (2) Comparison of denoising performance in colour image denoising under PTQ and QAT settings with other state-of-the-art FP32 models. (3) Comparison of accelerator throughput and energy efficiency with other denoiser models in the literature, where these metrics have been quantified. Experimental results demonstrate that our integrated solution not only introduces a high-throughput and high-energy-efficiency quantised denoising scheme into the NUKE workflow, but also achieves denoising performance comparable to, or surpassing, that of state-of-the-art FP32 and quantised models.

4.2 Grayscale Denoising Evaluation

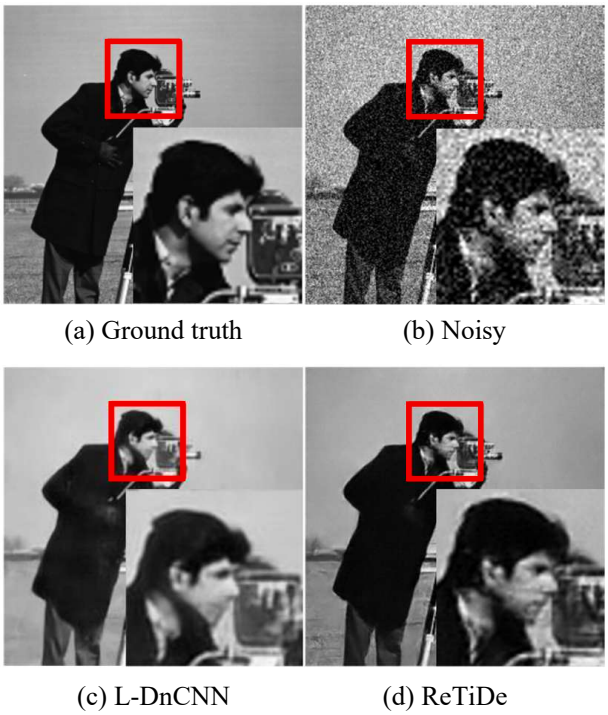
Grayscale denoising involves only single-channel input, which implies a reduction in the amount of input data available to the model. This naturally leads to differences in performance compared to colour denoising. To benchmark our model against existing denoising accelerators, we first evaluated denoising performance on grayscale images. Experiments were conducted on the classical BSD68 dataset and the higher-quality URBAN100 grayscale dataset, comparing our model against existing FP32 models, particularly FPGA-implemented quantised denoising models. For PSNR baseline tests in grayscale denoising, and to remain consistent with prior benchmarks, three noise levels with standard deviations of 15, 25, and 50 were employed. The experimental results are summarised in Table 1, with the PTQ and QAT models referred to as ReTiDe (P) and ReTiDe (Q) respectively. From the data, it can be observed that our quantised model achieves denoising performance close to that of 32-bit FP32 models under 8-bit quantisation. Moreover, in high-noise scenarios on the BSD68 dataset, our model outperforms the existing denoising accelerator L-DnCNN.

While PSNR reflects overall denoising performance, it does not capture all aspects of denoising; human visual perception and preservation of image details are equally important. The Set12 dataset is a classical baseline dataset; however, due to its early origin, the ground truth itself contains deviations from clean images. As shown in Figure 6(a), the ground truth images contain significant noise. Therefore, this dataset is primarily used for a detailed comparison of denoising results with other accelerators. Figure 6(b) shows images corrupted with Gaussian noise at a level of 35, and detailed comparisons are made between denoised images from the original L-DnCNN paper (c) and our quantised model (d). The results indicate that our model significantly outperforms L-DnCNN in detail preservation, particularly in facial features such as the mouth and hair contours, where the features are retained with higher accuracy and realism.

The superior detail-preserving capability of ReTiDe compared to L-DnCNN can be attributed to its encoder-decoder architecture with skip connections, which allows direct utilisation of low-level features for reconstructing high-resolution outputs. This enables the model to integrate high-level semantic information while preserving fine details. In contrast, DnCNN relies solely on deep convolutional stacking, which tends to smooth fine details across multiple abstraction layers.

Table 1: PSNR (*dB*) comparison of various algorithms for grayscale image denoising.

Method	BSD68 (gray)			URBAN100 (gray)		
	15	25	50	15	25	50
BM3D	30.95	25.32	24.89	31.91	29.06	24.45
FFDNet	31.45	28.96	25.16	33.76	31.41	28.09
IRCNN	31.46	28.79	25.11	33.08	29.62	24.53
DnCNN-20	31.60	29.14	26.20	33.76	30.19	19.34
SwinIR	31.76	29.10	25.40	33.44	30.43	25.47
ReTiDe	31.48	29.09	26.20	33.25	30.60	26.55
QNNs						
L-DnCNN	31.44	29.01	26.08	-	-	-
ReTiDe (P)	29.92	28.35	26.73	29.92	28.35	25.52
ReTiDe (Q)	30.94	29.23	26.73	30.20	28.46	25.61



(a) Ground truth (b) Noisy

(c) L-DnCNN (d) ReTiDe

Figure 6: Comparison of output results with other quantised image denoising models under the noise level of 35.

Table 2: PSNR (*dB*) and SSIM comparison of popular denoisers for colour denoising on BSD100.

Method	Blind/Nonblind	BSD100 (colour)									
		5		15		25		35		45	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
BM3D	Nonblind	39.85	0.98	33.17	0.9223	30.16	0.8598	28.17	0.8007	26.62	0.747
DnCNN	Blind	39.72	0.9728	33.46	0.9245	30.56	0.8711	28.62	0.8217	27.11	0.7766
FFDNet	Nonblind	39.84	0.9788	33.65	0.9265	31.00	0.8772	29.37	0.8337	28.23	0.7958
IRCNN	Nonblind	39.95	0.9789	33.41	0.9234	30.45	0.8678	28.43	0.8114	26.87	0.7578
ReTiDe (FP32)	Blind	39.46	0.9761	33.27	0.9205	30.65	0.8682	29.03	0.8224	27.89	0.7826
QNNs											
ReTiDe (P)	Blind	32.94	0.8943	30.38	0.8414	28.95	0.8149	27.96	0.7941	27.06	0.7713
ReTiDe (Q)	Blind	33.22	0.9425	31.03	0.9008	29.37	0.8576	28.14	0.8164	27.14	0.7811

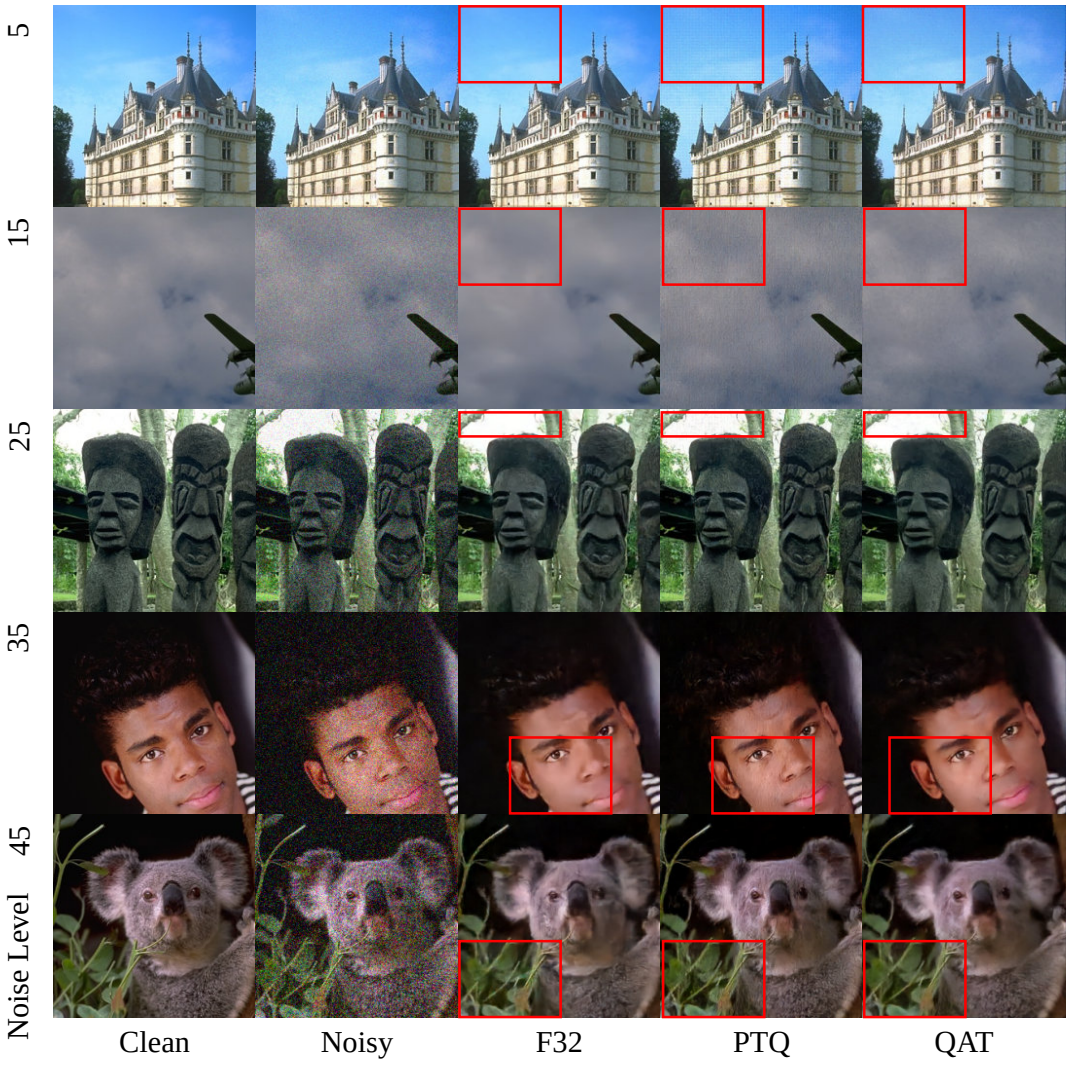


Figure 7: Denoising results of FP32 and quantised ReTiDe denoising models at different noise levels.

4.3 Colour Denoising Evaluation

In addition to PSNR performance comparisons and detail evaluations against existing baseline denoising accelerators, we further validate our quantised model ReTiDe on the colour dataset BSD100, which is closer to real-world application scenarios. We conduct baseline tests under Gaussian noise with finer-grained standard deviations of 5, 15, 25, 35, and 45, comparing our method with other state-of-the-art FP32 models. To better capture perceptual quality from a user perspective, we also introduce the SSIM as an evaluation metric. The experimental results are summarised in Table 2. Notably, under blind denoising, our model achieves performance comparable to or exceeding that of classical and advanced FP32 models, while QAT further improves both PSNR and SSIM metrics. In high-noise conditions, such as a noise level of 45, our 8-bit quantised model is only 1.09 dB and 0.0147 away from the best 32-bit FP32 model in terms of PSNR and SSIM, respectively. In contrast, under low-noise conditions, quantisation error becomes the dominant factor instead of noise, resulting in relatively larger performance gaps compared to FP32 models. We anticipate that hybrid-precision quantisation could mitigate this issue in future work, though such an exploration is beyond the scope of this paper.

To further assess the denoising detail preservation of our quantised model on colour images, we present results in Figure 7, showing clean, noisy, and denoised images under different noise levels for the FP32 model, the PTQ model, and the QAT model. We highlight regions with noticeable differences. Our observations show that quantisation errors and quantisation-induced noise introduced by PTQ can lead to inferior denoising performance, especially in relatively smooth background regions. However, QAT's quantisation-aware fine-tuning effectively alleviates this problem, achieving denoising quality nearly identical to that of the FP32 model. These denoising examples further demonstrate the robustness of our model on colour images across multiple noise levels.

4.4 Deployment Performance

We deployed the quantised model in the form of an `xmodel` on a server equipped with an Alveo U50 FPGA, while the corresponding FP32 models were deployed on a client equipped with an NVIDIA A4000 GPU and an Intel(R) Core(TM) Ultra 7 265K CPU. Batch denoising tasks were invoked via software Application Programming Interfaces (APIs) to evaluate the runtime throughput and energy efficiency of the quantised model. A one-minute warm-up inference was first performed to stabilise device operation, after which throughput was computed based on FPS and runtime performance. Power consumption was measured using the `powerstat` tool, by recording the average power difference between the IDLE state and active inference, and this value was adopted as the energy efficiency metric. The experimental results are summarised in Table 3.

The results demonstrate that our FPGA-based denoising inference achieves a throughput 37.71× higher than other FPGA-based baseline deep learning denoisers, reaching 3,746.09 GOPS. Compared to the 0.033s inference time of L-DnCNN [Kang et al. 2024], our model takes only 0.004s. This is attributed to our quantised inference optimisations tailored for multi-DPU architectures and multi-threaded parallel execution. Although TNet-mini [Tu et al. 2024] and L-DnCNN [Kang et al. 2024] achieve strong performance through the use of the Winograd algorithm and lightweight CNN structures, server-level FPGAs typically provide more efficient architectures, including high-bandwidth memory (HBM) and customizable DPUs. Moreover, Vitis-AI offers comprehensive system-level optimisations, such as operator fusion, layer-wise quantisation, and efficient memory scheduling, which significantly improve energy efficiency in practical deployment scenarios.

Table 3: Performance comparison across platforms: throughput, power, and energy efficiency.

Method	Platform	Thr. (GOPS)	Power (W)	Energy Eff. (GOPS/W)
L-DnCNN	I7-7700HQ CPU	29.5	45	0.66
TNet-mini	I5-12400F CPU	164.3	65	2.53
ReTiDe	U7-265K CPU	770.2	42.1	18.30
L-DnCNN	RTX 1070 GPU	1066.7	115	9.28
TNet-mini	RTX 2080Ti GPU	1785.7	250	7.14
ReTiDe	A4000 GPU	8,285.5	236.3	35.06
L-DnCNN	MZU03A-EG FPGA	41.8	2.4	17.18
TNet-mini	MZU03A-EG FPGA	99.3	2.6	38.51
ReTiDe	Alveo U50 FPGA	3,746.1	**22**	170.3

For energy efficiency, our approach surpassed baseline denoising accelerators by 4.42×, reaching 170.28 GOPS/W. These results indicate that our quantised denoising accelerator and its end-to-end deployment scheme deliver a significant advantage in processing large-scale denoising tasks with high energy efficiency compared to conventional hardware platforms, including CPUs, GPUs, and existing FPGA-based neural network denoising accelerators.

5 Conclusion

This work proposes a hardware-accelerated image denoising solution integrated into professional media processing software. The end-to-end framework offloads computationally intensive denoising tasks to a server-level FPGA DPU acceleration platform. Through multi-threading and multi-PE parallel quantised acceleration, the framework significantly improves real-time denoising throughput and energy efficiency. This approach bridges the gap between the advantages of state-of-the-art quantised denoising models in terms of real-time performance and energy efficiency, and the demands of professional image media processing software for handling computationally intensive media denoising tasks. For denoising performance, the proposed solution achieves results close to advanced FP32 models for both colour and grayscale images. Compared with existing FPGA-based deep learning denoising accelerators, it achieves 4.42× and 37.71× improvements in energy efficiency and throughput, respectively. This offers a new pathway for accelerating and offloading professional image processing algorithms. Future work will focus on further enhancing denoising detail performance using mixed-precision quantisation on more advanced models, exploring model sparsification to achieve even greater energy efficiency, and incorporating real-world image noise to improve the practical applicability of the proposed solution.

References

Eirikur Agustsson and Radu Timofte. 2017. NTIRE 2017 Challenge on Single Image Super-Resolution: Dataset and Study. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 1122–1131. doi:10.1109/CVPRW.2017.150

Carlo Beenakker and Christian Schönenberger. 2003. Quantum shot noise. *Physics Today* 56, 5 (2003), 37–42.

Clement Bled and Francois Pitie. 2022. Assessing advances in real noise image denoisers. In *Proceedings of the 19th ACM SIGGRAPH European Conference on Visual Media Production*. 1–9.

Clément Bled and François Pitié. 2024. Lightweight video denoising using a classic Bayesian backbone. In *2024 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 1–6.

Jonas Brenig and Radu Timofte. 2025. Higher fidelity perceptual image and video compression with a latent conditioned residual denoising diffusion Model. In *Computer Vision – ECCV 2024 Workshops*, Alessio Del Bue, Cristian Canton, Jordi Pont-Tuset, and Tatiana Tommasi (Eds.). Springer Nature Switzerland, Cham, 194–210.

Antoni Buades, Bartomeu Coll, and J-M Morel. 2005. A non-local algorithm for image denoising. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, Vol. 2. IEEE, 60–65.

Patrick L Combettes and Jean-Christophe Pesquet. 2004. Wavelet-constrained image restoration. *International Journal of Wavelets, Multiresolution and Information Processing* 2, 04 (2004), 371–389.

Pierrick Coupé, Pierre Yger, Sylvain Prima, Pierre Hellier, Charles Kervrann, and Christian Barillot. 2008. An optimized blockwise nonlocal means denoising filter for 3-D magnetic resonance images. *IEEE transactions on medical imaging* 27, 4 (2008), 425–441.

Swapnil Deelip Dabhade, GN Rathna, and Kunal Narayan Chaudhury. 2017. A reconfigurable and scalable FPGA architecture for bilateral filtering. *IEEE Transactions on Industrial Electronics* 65, 2 (2017), 1459–1469.

Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. 2007. Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Transactions on image processing* 16, 8 (2007), 2080–2095.

Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. 2009. BM3D Image Denoising with Shape-Adaptive Principal Component Analysis. In *SPARS'09 - Signal Processing with Adaptive Sparse Structured Representations*, Rémi Gribonval (Ed.). Inria Rennes - Bretagne Atlantique, Saint Malo, France. https://inria.hal.science/inria-00369582

Basar Demir, Yikang Liu, Xiao Chen, Eric Z Chen, Lin Zhao, Boris Mailhe, Terrence Chen, and Shanhui Sun. 2025. DiffDenoise: self-supervised medical image denoising with conditional diffusion models. *arXiv preprint arXiv:2504.00264* (2025).

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929* (2020).

Alliance for Open Media. n.d.. AV1 Codec source code repository. https://aomedia.googlesource.com/aom. Accessed: 2024-05-21.

Anna Gabiger-Rose, Matthias Kube, Robert Weigel, and Richard Rose. 2013. An FPGA-based fully synchronized design of a bilateral filter for real-time image denoising. *IEEE Transactions on Industrial Electronics* 61, 8 (2013), 4093–4104.

Guy Gilboa and Stanley Osher. 2009. Nonlocal operators with applications to image processing. *Multiscale Modeling & Simulation* 7, 3 (2009), 1005–1028.

Neat Video GmbH. 2024. *Neat Video*. https://www.neatvideo.com/ Accessed: 2024-10-09.

Shi Guo, Zifei Yan, Kai Zhang, Wangmeng Zuo, and Lei Zhang. 2019. Toward convolutional blind denoising of real photographs. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 1712–1722.

Javier Gurrola-Ramos, Oscar Dalmau, and Teresa E Alarcón. 2021. A residual dense U-Net neural network for image denoising. *IEEE Access* 9 (2021), 31742–31754.

Song Han, Huizi Mao, and William J. Dally. 2016. Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding. arXiv:1510.00149 [cs.CV] https://arxiv.org/abs/1510.00149

Mattias P Heinrich, Maik Stille, and Thorsten M Buzug. 2018. Residual U-Net convolutional neural network architecture for low-dose CT denoising. *Current Directions in Biomedical Engineering* 4, 1 (2018), 297–300.

Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. 2015. Single image super-resolution from transformed self-exemplars. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 5197–5206.

Dileepan Joseph and Steve Collins. 2001. Modelling, calibration and correction of nonlinear illumination dependent fixed pattern noise in logarithmic CMOS image sensors. In *IMTC 2001. Proceedings of the 18th IEEE Instrumentation and Measurement Technology Conference. Rediscovering Measurement in the Age of Informatics (Cat. No. 01CH 37188)*, Vol. 2. IEEE, 1296–1301.

S Kala, Babita R Jose, Jimson Mathew, and S Nalesh. 2019. High-performance CNN accelerator on FPGA using unified winograd-GEMM architecture. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 27, 12 (2019), 2816–2828.

Wenzheng Kang, Xinchun Wu, Ming Zhang, Xiaojun Zhang, Xiaobing Huang, Biao Sun, and Qianneng Zhou. 2024. Improvement and Hardware Design of Image Denoising Algorithm Based on Deep Learning. In *2024 9th International Conference on Integrated Circuits and Microsystems (ICICM)*. IEEE, 671–676.

Salman Khan, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, and Mubarak Shah. 2022. Transformers in vision: A survey. *ACM computing surveys (CSUR)* 54, 10s (2022), 1–41.

Yawei Li, Kai Zhang, Jingyun Liang, Jiezhang Cao, Ce Liu, Rui Gong, Yulun Zhang, Hao Tang, Yun Liu, Denis Demandolx, et al. 2023. Lsdir: A large scale dataset for image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1775–1787.

Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. 2021. SwinIR: Image Restoration Using Swin Transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 1833–1844.

Pengju Liu, Hongzhi Zhang, Kai Zhang, Liang Lin, and Wangmeng Zuo. 2018. Multi-level Wavelet-CNN for Image Restoration. *arXiv:1805.07071 [cs]* (May 2018). http://arxiv.org/abs/1805.07071 arXiv: 1805.07071 version: 2.

Xinqiao Liu and Abbas El Gamal. 2001. Photocurrent estimation from multiple non-destructive samples in CMOS image sensor. In *Sensors and Camera Systems for Scientific, Industrial, and Digital Photography Applications II*, Vol. 4306. SPIE, 450–458. doi:10.1117/12.426983

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 10012–10022.

The Foundry Visionmongers Ltd. 2023. *Nuke*. https://www.foundry.com/products/nuke Accessed: 2024-10-09.

Mona Mahmoudi and Guillermo Sapiro. 2005. Fast image and video denoising via nonlocal means of similar neighborhoods. *IEEE signal processing letters* 12, 12 (2005), 839–842.

Maurits Malfait and Dirk Roose. 1997. Wavelet-based image denoising using a Markov random field a priori model. *IEEE Transactions on image processing* 6, 4 (1997), 549–565.

Stephane G Mallat. 1989. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE transactions on pattern analysis and machine intelligence* 11, 7 (1989), 674–693.

David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. 2001. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings eighth IEEE international conference on computer vision. ICCV 2001*, Vol. 2. IEEE, 416–423.

Emmet Murphy, Shashwat Khandelwal, and Shreejith Shanker. 2023. Custom precision accelerators for energy-efficient image-to-image transformations in motion picture workflows. In *Applications of Digital Image Processing XLVI*, Vol. 12674. SPIE, 191–202.

Xuebin Qin, Zichen Zhang, Chenyang Huang, Masood Dehghan, Osmar R Zaiane, and Martin Jagersand. 2020. U2-Net: Going deeper with nested U-structure for salient object detection. *Pattern recognition* 106 (2020), 107404.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-Net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*. Springer, 234–241.

Walter Schottky. 1918. Über spontane Stromschwankungen in verschiedenen Elektrizitätsleitern. *Annalen der physik* 362, 23 (1918), 541–567.

Walter Schottky. 2018. On spontaneous current fluctuations in various electrical conductors. *Journal of Micro/Nanolithography, MEMS, and MOEMS* 17, 4 (2018), 041001–041001.

Fanny Spagnolo, Pasquale Corsonello, Fabio Frustaci, and Stefania Perri. 2023. Design of approximate bilateral filters for image denoising on FPGAs. *IEEE Access* 11 (2023), 1990–2000.

Fanny Spagnolo, Pasquale Corsonello, Fabio Frustaci, and Stefania Perri. 2024. Approximate bilateral filters for real-time and low-energy imaging applications on FPGAs. *The Journal of Supercomputing* 80, 11 (2024), 15894–15916.

The Foundry Visionmongers Limited. 2025. Cattery: A library of open-source machine learning models for Nuke. Foundry Community website, Nuke section. https://community.foundry.com/cattery Accessed on 21 August 2025; "a library of open-source machine learning models converted to .cat files to run natively inside Nuke.".

Radu Timofte, Eirikur Agustsson, Luc Van Gool, Ming-Hsuan Yang, Lei Zhang, Bee Lim, et al. 2017. NTIRE 2017 Challenge on Single Image Super-Resolution: Methods and Results. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Linfeng Tu, Yuliang Gu, Xiaobing Huang, and Xinchun Wu. 2024. Lightweight Design of Image Denoising based on TNet mini. In *2024 4th International Symposium on Artificial Intelligence and Intelligent Manufacturing (AIIM)*. IEEE, 646–649.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

Jin Wang, Yanwen Guo, Yiting Ying, Yanli Liu, and Qunsheng Peng. 2006. Fast non-local algorithm for image denoising. In *2006 International Conference on Image Processing*. IEEE, 1429–1432.

Shuo-Fei Wang, Wen-Kai Yu, and Ya-Xin Li. 2020. Multi-wavelet residual dense convolutional neural network for image denoising. *IEEE Access* 8 (2020), 214413–214424.

Xijun Wang, Prateek Chennuri, Yu Yuan, Bole Ma, Xingguang Zhang, and Stanley Chan. 2024. Personalized Generative Low-light Image Denoising and Enhancement. *arXiv preprint arXiv:2412.14327* (2024).

Zhendong Wang, Xiaodong Cun, Jianmin Bao, Wengang Zhou, Jianzhuang Liu, and Houqiang Li. 2022. Uformer: A general u-shaped transformer for image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 17683–17693.

JiaBao Wen, Yan Feng, and ZhiQiang Li. 2024. A high-throughput hardware architecture for bilateral filter with configurable convolution and cost-effective MAC unit. *IEICE Electronics Express* 21, 13 (2024), 20240276–20240276.

Ailin Xie, Ao Zhang, and Guohui Mei. 2024. An FPGA-based hardware architecture of gaussian-adaptive bilateral filter for real-time image denoising. *IEEE Access* (2024).

David X. D. Yang and Abbas El Gamal. 1999. Comparative analysis of SNR for image sensors with enhanced dynamic range. In *Sensors, Cameras, and Systems for Scientific/Industrial Applications*, Vol. 3649. SPIE, 197–211. doi:10.1117/12.347075

Ruoheng Yao, Lei Chen, Pingcheng Dong, Zhuoyu Chen, and Fengwei An. 2022. A compact hardware architecture for bilateral filter with the combination of approximate computing and look-up table. *IEEE Transactions on Circuits and Systems II: Express Briefs* 69, 7 (2022), 3324–3328.

Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. 2022. Restormer: Efficient Transformer for high-resolution image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 5728–5739.

Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. 2021. Multi-stage progressive image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 14821–14831.

Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. 2017a. Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising. *IEEE Transactions on Image Processing* 26, 7 (July 2017), 3142–3155. doi:10.1109/TIP.2017.2662206 arXiv: 1608.03981 version: 1.

Kai Zhang, Wangmeng Zuo, Shuhang Gu, and Lei Zhang. 2017b. Learning Deep CNN Denoiser Prior for Image Restoration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Kai Zhang, Wangmeng Zuo, and Lei Zhang. 2018. FFDNet: Toward a Fast and Flexible Solution for CNN based Image Denoising. *IEEE Transactions on Image Processing* (2018).

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*. 2223–2232.