

Development of a Super-Resolution Satellite Image Processing System on an Edge AI Platform

Kanapoj Ngambenjavitkul
Multimedia Data Analytic and
Processing Research Unit,
Department of Electrical Engineering,
Chulalongkorn University
Bangkok, Thailand
6670031721@student.chula.ac.th

Amir Hajian
Multimedia Data Analytic and
Processing Research Unit,
Department of Electrical Engineering,
Chulalongkorn University
Bangkok, Thailand
amirhajian85@gmail.com

Watchara Ruangsang
King Mongkut's University
of Technology Thonburi,
Bangkok, Thailand
watchara.ruan@kmutt.ac.th

Supavadee Aramvith
Multimedia Data Analytic and
Processing Research Unit,
Department of Electrical Engineering,
Chulalongkorn University
Bangkok, Thailand
supavadee.a@chula.ac.th

Abstract— High-resolution satellite imagery is essential for many remote sensing applications, such as land-use mapping, infrastructure monitoring, and environmental assessment. Unfortunately, acquiring high-resolution data is expensive, constrained by limited bandwidth, and limited by infrequent satellite passes. Super-resolution (SR) techniques offer a software-based approach to enhance image resolution by reconstructing high-resolution images from low-resolution inputs, utilizing deep learning. In this work, we present a fully implemented, lightweight attention-enhanced U-Net model for real-time satellite image super-resolution on an edge AI platform. We optimize the model using the Xilinx Vitis AI toolchain and deploy it on the Xilinx Kria KV260 FPGA board. The development pipeline includes quantization-aware training, pruning, model compilation, and hardware deployment. The network architecture is carefully designed to preserve spatial detail while being compatible with quantization and FPGA constraints. We evaluated the system on the UCMerced Land Use dataset with $\times 2$ and $\times 4$ upscaling factors. Results show that the INT8-quantized model produces high-quality super-resolved images with low inference latency and efficient resource usage. These outcomes demonstrate the practicality of the proposed system for real-time, embedded remote sensing applications.

Keywords—Super-resolution, remote sensing, edge AI, quantization aware training, Xilinx Kria KV260, Vitis AI, FPGA deployment, U-Net

I. INTRODUCTION

High-resolution satellite imagery is crucial for various remote sensing applications, including land-use mapping, infrastructure monitoring, and environmental assessment. However, acquiring high-resolution data remains challenging due to limitations in cost, bandwidth, and revisit frequency. These limitations reduce the availability of timely and detailed imagery, which can hinder tasks that require fine-grained spatial information such as edge detection, texture analysis, and structural mapping.

Super-resolution (SR) techniques have emerged as a cost-effective solution by reconstructing high-resolution images from low-resolution inputs using computational methods. In remote sensing, SR enables enhanced image interpretation without requiring additional data acquisition or hardware upgrades. Deep learning models, particularly those based on convolutional neural networks (CNNs), have demonstrated remarkable success in this domain by learning complex mappings from large-scale training data for a GPU-based environment. However, these models are often too large or

computationally demanding for real-time or edge deployment.

Edge AI platforms such as the Xilinx Kria KV260 offer a promising solution by enabling low-power, high-performance inference on embedded systems. The platform integrates ARM processors, FPGA fabric, and a Deep Processing Unit (DPU), and supports optimization workflows using the Vitis AI toolchain. These features make it suitable for deploying lightweight deep learning models in field settings such as UAVs or remote monitoring stations.

To be deployable on edge hardware, SR models must be efficient, quantization-friendly, and accurate enough to recover fine textures and small objects. While several recent models have pushed the boundaries of accuracy in Remote Sensing image Super Resolution (RSISR), many are still too large or impractical for embedded use. This motivates the need for models that strike a balance between accuracy, model size, and hardware compatibility.

This paper presents a complete, edge-deployable SR system optimized for the Xilinx Kria KV260. Our work builds upon the AERU-Net [3], a 2025 U-Net-based model that integrates edge recovery and attention mechanisms to enhance reconstruction quality while reducing model parameters. We further optimize the model for quantization-aware training and real-time deployment.

II. RELATED WORKS

A. Deep Learning based Single Image Super Resolution

Single Image Super-Resolution (SISR) plays a critical role in remote sensing by reconstructing high-resolution (HR) imagery from low-resolution (LR) inputs. The enhanced spatial resolution significantly improves downstream applications such as land classification, infrastructure detection, and environmental analysis. Traditional interpolation methods, although computationally efficient, often fail to recover fine structures and texture details, particularly in complex scenes. The advent of deep learning has transformed the SISR domain. SRCNN [1] was one of the earliest convolutional models to learn end-to-end mappings for super-resolution, laying the groundwork for deeper architectures. It was followed by deeper models such as EDSR [2] and RCAN [4], which introduced residual learning and channel attention, respectively, to improve reconstruction quality. However, their high computational complexity and

large number of parameters present significant obstacles for real-time or edge-based applications.

To overcome these limitations, researchers have developed models specifically targeting Remote Sensing Image Super-Resolution (RSISR). U-Net and its derivatives have gained prominence due to their encoder-decoder architecture and skip connections, which facilitate multi-scale context fusion while preserving spatial detail. Models based on U-Net have demonstrated effectiveness in recovering small-scale structures and sharp edges in satellite images. Recent advances include the integration of transformer-based modules and attention mechanisms into RSISR networks. TransENet [5] introduced a multistage enhancement structure that combines high- and low-dimensional feature embeddings. Although it achieved competitive performance, its large parameter counts and inefficient upsampling layers limited its practicality, particularly on resource-constrained hardware.

In 2024, CSA-FE [6] further developed this approach by incorporating channel and spatial attention modules to enhance feature discrimination and improve high-frequency detail restoration. Despite improvements in visual quality, CSA-FE suffered from high computational complexity and limited ability to preserve content-aware attention across different scales.

HAUNet [7] further advanced the U-Net architecture by embedding hybrid attention modules to capture global spatial context and content information. It also introduced a cross-scale interaction module to better link encoder outputs at different resolution levels. While HAUNet achieved strong performance in capturing abstract semantics, it demonstrated limited capability in recovering sharp edges and was computationally demanding.

To address these challenges, AERU-Net [3], was proposed in 2025 as a lightweight and deployable RSISR model. It enhances the standard U-Net backbone by introducing three core components: an Edge Recovery Block (ERB) for structural fidelity, spatial and channel attention modules (SAM and CAM) for adaptive focus, and a Cross-Scale Interaction Module (CIM) for efficient multi-level communication. These modules collectively improve both reconstruction accuracy and hardware compatibility.

This study builds upon AERU-Net by modifying its architecture and training procedures to support quantization-aware training (QAT). The resulting model is optimized for deployment on the Xilinx Kria KV260 platform, enabling real-time inference with reduced memory and compute overhead.

B. Xilinx Kria KV260 board

Edge AI platforms have emerged as a practical solution for deploying deep learning models in resource-constrained environments, such as those found in UAVs, field stations, and remote sensing payloads. These platforms aim to provide efficient, low-latency inference with limited power and memory resources. Among these, the Xilinx Kria KV260 stands out as a reconfigurable System-on-Chip (SoC) platform that strikes a balance between performance,

flexibility, and energy efficiency. The KV260 is built on the Zynq UltraScale+ MPSoC architecture, which combines a quad-core ARM Cortex-A53 processor, a dual-core Cortex-R5 real-time processor, and programmable logic based on FPGA fabric [8]. This design allows heavy AI computations to be offloaded to hardware while general processing runs on the ARM cores. Figure 1 shows the KV260 development board, which pairs the Zynq MPSoC module with a carrier board for vision applications.

A key feature of the KV260 platform is its integrated Deep Processing Unit, a specialized neural network accelerator implemented in the FPGA logic. The DPU is an IP core optimized for deep learning workloads, particularly convolutional neural networks, leveraging parallel computation and fast on-chip memory to achieve high throughput. In essence, the DPU is a dedicated AI engine for convolutional network inference, freeing the ARM cores from the bulk of the math-intensive work. Xilinx provides a comprehensive software stack called Vitis AI to support deploying models onto the DPU.



Fig. 1. Xilinx Kria KV260 board

Using the Vitis AI toolchain, neural network models trained in frameworks such as PyTorch or TensorFlow can be converted into a DPU-compatible format through several optimization steps. These include quantization, pruning, and compilation into a hardware executable. Quantization is especially crucial. By reducing model precision from 32-bit floating point to 8-bit integer, one can shrink the model size by a factor of four and often achieve a two- to four-times speedup in inference, with significantly lower memory bandwidth usage. This int8 optimization yields faster, smaller models while maintaining near-original accuracy when done with proper calibration or quantization-aware training. Pruning further trims the model by removing redundant weights, which also helps meet the limited on-chip memory constraints.

III. METHODOLOGY

A. Network Architecture

Our proposed network builds upon the AERU-Net architecture, which is specifically designed for remote sensing image super-resolution. AERU-Net follows the standard U-Net design, which uses an encoder-decoder structure with skip connections. This structure helps preserve spatial details and allows the network to extract features at multiple scales. The model starts with a 3×3 convolution to extract shallow features from the low-resolution input image. These features are then passed through three encoder stages. Each stage reduces the spatial

resolution while increasing the number of channels, allowing the network to learn both low- and high-level information. The outputs from the three encoder levels, called F_1 , F_2 , F_3 are combined using a Cross-Scale Interaction Module (CIM), which merges information across different feature scales. The CIM outputs are denoted as OUT_1 , OUT_2 , OUT_3 .

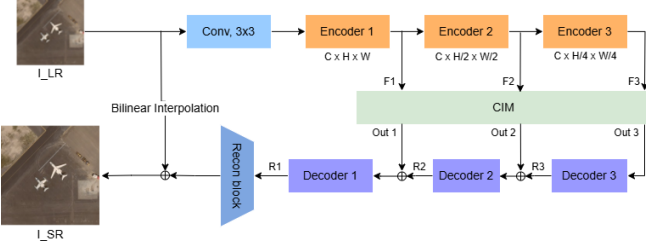


Fig. 2. Overview of the Algorithm

The decoder path mirrors the encoder. It takes the CIM outputs and progressively upsamples them, combining each decoder output with the matching encoder features through skip connections. This helps the network recover fine textures and object boundaries. The final decoder result is fused with a bilinearly upsampled version of the input using a reconstruction block. This improves edge sharpness and overall consistency. The whole process is expressed as:

$$R_1 = \text{Decoder}_1(OUT_1 \oplus \text{Decoder}_2(OUT_2 \oplus \text{Decoder}_3(OUT_3))) \quad (1)$$

$$I_{SR} = \text{Recon}(R_1) \oplus \text{BI}(I_{LR}) \quad (2)$$

where I_{SR} is the super-resolved output, I_{LR} is the input, and $\text{BI}(I_{LR})$ is its bilinear interpolation. The \oplus symbol indicates element-wise addition or concatenation

To enhance feature learning and improve the reconstruction of fine spatial details, AERU-Net incorporates two specialized attention mechanisms: the Channel Attention Module (CAM) and the Spatial Attention Module (SAM) [9]. The CAM helps the network learn the most relevant channels for specific features. It emphasizes texture channels when identifying urban types and prioritizes intensity channels when distinguishing between asphalt roads and concrete buildings.

This enables the network to selectively amplify channel-wise information that is most indicative of structural and material variations in remote sensing imagery. The SAM helps the network identify which spatial regions of the image require more attention during processing. In AERU-Net, both CAM and SAM are integrated at the first level of the encoder and decoder, where spatial resolution is high and local structures are more apparent. At deeper levels, only CAM is retained due to the increased abstraction of features and reduced spatial resolution.

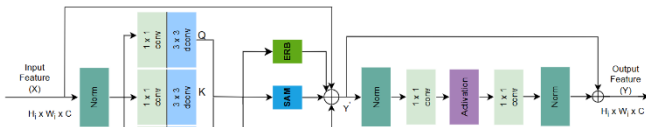


Fig. 3. Encoder 1, ERB, SAM, CAM, FFN

In addition to attention mechanisms, AERU-Net incorporates Edge Recovery Blocks (ERBs) throughout all encoder and decoder stages. The ERB enhances the model's ability to capture high-frequency details by applying classical edge-detection filters, such as Sobel and Laplacian operators, to extract directional and multi-scale edge information. This module is particularly effective in reinforcing boundary representations between land cover classes, such as building footprints, roads, coastlines, and field edges. The ERB outputs are fused with attention-based features through residual connections, which enhance learning stability and preserve detailed spatial structures. In the optimized version of the model, the ERB is simplified to reduce computational cost while maintaining effective edge enhancement. The proposed architecture builds upon AERU-Net with structural simplifications aimed at improving deployment efficiency and quantization compatibility.

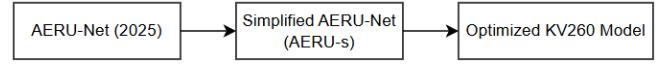


Fig. 4. Model Overview

Figure 4 illustrates the model lineage from the AERU-Net to our final optimized KV260. The AERU-s refers to a partially simplified AERU-Net variant trained in FP32. The final optimized model is a further compressed version designed for INT8 quantization and deployment on the Xilinx Kria KV260.

B. Model Optimization

Deploying the above model on the FPGA-based DPU required a series of adaptations to reduce complexity and ensure compatibility with the Vitis AI toolchain. We modified both the architecture and the training process to create an optimized version of the network that runs efficiently in 8-bit integer format on the KV260.

Deployment constraints, including the need for operator support on the DPU and the requirement for quantization-aware operations, guided the optimization process. The baseline model utilized advanced normalization layers, including LayerNorm and GroupNorm, in addition to BatchNorm. These were removed, keeping only standard Batch Normalization layers which are well-supported on the DPU. This not only improves compatibility but also reduces computational overhead.

The attention pathway also requires simplification. The original design integrated three modules: CAM, SAM, and an ERB. In the optimized model, the CAM and SAM modules were retained but stripped of normalization layers and redesigned to rely solely on quantization-friendly pooling and pointwise operations. The edge attention block was excluded entirely due to its complexity and the lack of support from operators. To reduce computation without compromising representation capacity, we replaced the original feed-forward network (FFN) [10] structure with a lightweight configuration using SimpleGate activation combined with depthwise separable convolutions. This design allows efficient elementwise nonlinearity and spatial filtering while reducing multiply-accumulate operations. Similarly, tensor rearrangement operations previously

dependent on the einops library and custom projection heads were re-implemented using `.view()` and `.permute()` functions to meet the operator constraints of Vitis AI. The output reconstruction block was also modified. Bilinear upsampling and residual fusion in the model were replaced with an efficient pixel shuffle operation, which offers a hardware-friendly alternative for spatial resolution expansion.

C. Deployment Workflow Using Vitis AI

Deploying the quantized super-resolution model on the Kria KV260 involves several stages, using the Xilinx Vitis AI toolchain to transition from a trained network to efficient on-board inference.

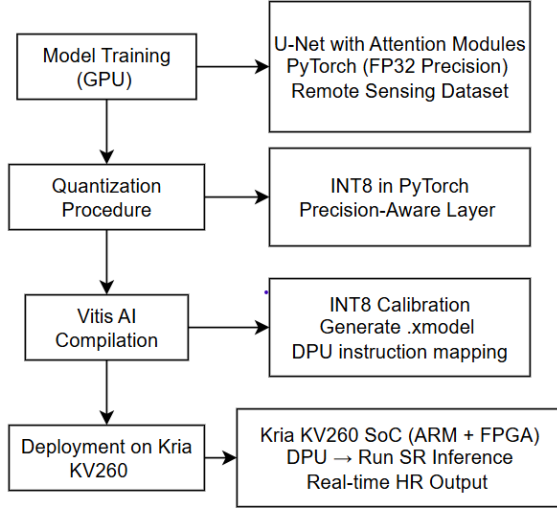


Fig. 5. Overview of the Optimization and Deployment Workflow on the Vitis AI tool

We first trained the model on a GPU using PyTorch in full precision to establish a high-performance baseline. After the initial convergence, we applied quantization-aware training to fine-tune the network. During this phase, fake quantization layers simulated 8-bit arithmetic for weights and activations during the forward pass. The model progressively adjusted to these constraints, resulting in a set of weights that were robust to low-precision inference. Once training was complete, the model was exported to the ONNX format, which is compatible with the Xilinx toolchain.

The ONNX model was passed through the Vitis AI Quantizer, which statically quantified weights and activations to INT8. We enabled bias correction to adjust for quantization bias and ensured that all quantizable layers met the constraints of the DPU runtime. Unsupported operations were eliminated during model design to avoid post-quantization incompatibilities.

After quantization, the model was compiled using the Vitis AI Compiler targeting the DPUCZDX8G architecture on the Kria KV260. This step generated a `.xmodel` binary file, containing the instructions required for the Deep Processing Unit (DPU) to perform inference. The `.xmodel` file was deployed onto the board along with a Python wrapper for data preprocessing and postprocessing. Fig. 4 illustrates the full deployment pipeline. The host system handles image input and preprocessing, forwards the image to the KV260 over USB or Ethernet, and invokes the DPU via Vitis AI Runtime (VART). The DPU executes the

quantized model, returning an upscaled image. Postprocessing, including image clipping and format conversion, is performed on the host side before the output is displayed or stored.

This deployment workflow ensures efficient utilization of the KV260 hardware resources. The entire system can perform INT8 super-resolution inference at real-time speeds with significantly reduced memory and energy consumption, making it suitable for edge scenarios such as UAV image processing and field-based analytics.

IV. EXPERIMENT AND RESULT

A. Datasets

We evaluated our model on the UCMerced Land Use dataset [11], which comprises 2,100 aerial RGB images across 21 scene types, including agricultural, residential, and industrial areas. Each image is 256×256 pixels with a 0.3-meter resolution, making it ideal for testing spatial enhancement. The dataset was split into 70% training, 15% validation, and 15% testing. To simulate low-resolution inputs, we applied bicubic downsampling and tested the model with $\times 2$ and $\times 4$ upscaling.

B. Experiment setup

The training and quantization were performed on a local workstation with an Intel Core i7-14700 CPU, 32 GB RAM, and an NVIDIA RTX 4090 GPU. We train a DPU-optimized super-resolution model on the UCMerced dataset, RGB aerial images, for two upscaling factors $\times 2$ and $\times 4$. This optimized model has significantly fewer parameters than the AERU Net model, allowing it to meet FPGA deployment constraints. The parameter count is reduced to approximately 0.29 M, compared to 0.9 M in the AERU-s. The reduction is achieved by using a slimmer architecture with fewer feature channels and blocks while maintaining performance.

We train each model for 300 epochs on the UCMerced training set using 1050 images and a batch size of 4 for $\times 4$ and 8 for $\times 2$. The Adam optimizer is used with a learning rate of 1×10^{-4} . The loss function combines L1 and L2 pixel losses with a fast Fourier transform loss, weighted at 0.1, to help preserve textural details. Validation performance is monitored on 105 images. Early stopping is set with a patience of 30 epochs to prevent overfitting. After training, the float model achieves a peak PSNR [12] of approximately 32.6 dB for $\times 2$ and 26.7 dB for $\times 4$ on the validation set, utilizing the optimized architecture.

Following float training, the model is quantized to 8-bit format using the Xilinx Vitis AI toolchain. Post-training calibration is first applied on a small subset of images to initialize quantization parameters. Then, quantization-aware training is conducted for 150 additional epochs with a reduced learning rate of 5×10^{-6} . During this stage, weights and activations are constrained to the INT8 data type. Initially, PSNR drops, for example, from 26.7 to 25.3 dB for $\times 4$ after calibration, and further to 23.9 dB during early epochs. By epoch 150, PSNR recovers to approximately 24.3 to 24.5 dB. A similar pattern is observed for the $\times 2$ model, where PSNR drops from 32.5 to 29.3 dB and later recovers to 29.8 dB.

Final quantized models are exported to the XModel format and compiled using Vitis AI. The entire pipeline is implemented in PyTorch with quantization and deployment handled through the Vitis AI toolchain. The deployed XModel runs on the Xilinx Kria KV260 FPGA with a deep processing unit accelerator. Vitis AI Runtime is used for inference. Input patches of size 32×32 for $\times 4$ or 64×64 for $\times 2$ are upsampled to 128×128 . These are tiled to reconstruct full images due to on-chip memory constraints. The result is a fully integer 8-bit model compatible with DPU execution and suitable for real-time edge inference.

C. Performance on Kria KV260

In this section, we present the performance results on the Kria KV260, comparing AERU-Net with our optimized model for $2\times$ and $4\times$ super-resolution on the Xilinx KV260.

At $\times 2$ upscaling, AERU-Net achieves a higher PSNR (34.64 dB) than our FP32 model (32.56 dB), although their SSIM values are nearly identical (~ 0.93). AERU-Net also attains a better spectral correlation and a lower SAM error, indicating it preserves spectral content slightly more faithfully. Our INT8 quantized model exhibits a noticeable drop in quality at $\times 2$, with PSNR falling to 29.72 dB and SSIM to 0.8698, accompanied by reduced SCC (0.3606) and a higher SAM (0.0691) compared to the FP32 version. This quantization impact reflects the trade-off for efficiency, as the INT8 model runs faster on the KV260 but with degraded image fidelity.

At $\times 4$ upscaling, the gap between AERU-Net and our FP32 model narrows in some respects. AERU-Net's PSNR is approximately 1.3 dB higher, but our model slightly outperforms in SSIM, suggesting it retains structural details well, even at high magnification. AERU-Net still maintains an advantage in spectral accuracy with a higher SCC and a marginally lower SAM. Once again, the INT8 struggles with this larger scale factor, as its PSNR drops to 24.59 dB and SSIM to 0.6856, alongside a significantly lower SCC and higher SAM.

Despite the slight performance loss, our model's strength lies in its extreme efficiency.

TABLE I. PERFORMANCE OF OPTIMIZED MODEL-KV260 ON UCMERCE DATASET

<i>Model</i>	<i>PSNR</i>	<i>SSIM</i>	<i>SCC</i>	<i>SAM</i>
AERU-Net ($\times 2$)	34.64	0.9346	0.6514	0.0478
Our ($\times 2$, FP32)	32.56	0.9332	0.0593	0.0539
Our ($\times 2$, INT8)	29.72	0.8698	0.3606	0.0691
AERU-Net ($\times 4$)	27.97	0.7686	0.2842	0.1007
Our ($\times 4$, FP32)	26.69	0.7939	0.2172	0.1020
Our ($\times 4$, INT8)	24.59	0.6856	0.0976	0.1231

Higher values are better for PSNR, SSIM, and SCC [12]. Lower values are better for SAM. End-to-end throughput includes all memory and processing overheads. As shown, the INT8 models on KV260 produce results close to the float baseline.

In addition to speed, memory footprint is significantly reduced. Table II compares model sizes. The AERU-s model

for $\times 4$ has 0.9260 million parameters and requires approximately 3.70 megabytes. The optimized version has 0.2952 million parameters and a size of 1.18 megabytes in FP32. After INT8 quantization, this reduces to 0.30 megabytes. The result is more than a 12 times reduction in size. The model fits within the KV260 SRAM, avoiding the need for external DDR memory and improving both latency and energy efficiency.

TABLE II. PERFORMANCE OF AERU-NET VS. OUR FP32/INT8 MODELS ON THE KRIA KV260 ($2\times$ AND $4\times$ SR)

<i>Model</i>	<i>Params (M)</i>
AERU-Net ($\times 4$)	0.706
AERU-S ($\times 4$)	0.9260
Optimized KV260 INT8 ($\times 4$)	0.2952
AERU-Net ($\times 2$)	0.705
AERU-S ($\times 2$)	0.9052
Optimized KV260 INT8 ($\times 2$)	0.2744

Float models are estimated to be using 4 bytes per parameter. The INT8 models use 1 byte per parameter. The optimized models contain approximately 70 percent fewer parameters, and quantization further reduces the memory by 75 percent.

TABLE III. COMPARISON ON UCMERCE DATASET AT SCALE OF $\times 4$.

<i>Model</i>	<i>Parameters (M)</i>
TransENet	37.46
CSA-FE	32.35
HAUNet	1.62
AERU-Net	0.706
Optimized KV260	0.2952

As shown in the table above, the optimized KV260 model contains only 0.295 million parameters. This is less than half the size of AERU-Net at 0.706 million and significantly smaller than other models such as TransENet and CSA-FE. Its compact architecture makes it highly suitable for deployment on the resource-constrained KV260 platform, providing faster inference and lower memory usage. In practice, the proposed model sacrifices only a small amount of accuracy to achieve substantial efficiency gains.

Visual outputs confirm that the optimized model produces perceptually sharp and structurally consistent results.

1) $\times 2$ upscale



Fig. 6. “building52” from the UCMerced dataset: GT (Left), LR (Middle), and Our Optimized KV260 Result (Right)



Fig. 7. “airplane80” from the UCMerced dataset: GT (Left), LR (Middle), and Our Optimized KV260 Result (Right)

Figure 6 and Figure 7 showcase the output from our deployed INT8 model at $\times 2$ upscaling on example scenes from the UCMerced dataset. While slight softening is observed compared to the ground truth, fine details such as rooftop edges and airplane contours are successfully recovered

2) $\times 4$ upscale



Fig. 8. “runway65” from the UCMerced dataset: GT (Left), LR (Middle), and Our Optimized KV260 Result (Right)



Fig. 9. “storagetanks81” from the UCMerced dataset: GT (Left), LR (Middle), and Our Optimized KV260 Result (Right)

Figures 8 and 9 present results at the more challenging $\times 4$ upscaling. Despite the increased difficulty, the model successfully recovers key structures such as the circular outline of storage tanks and the linear geometry of runways, although textures remain partially blurred and some edge artifacts are visible.

V. CONCLUSION

This paper presents a complete framework for deploying a deep learning-based super-resolution model for remote sensing imagery on edge devices. The proposed system is derived from a structurally simplified version of AERU-Net, adapted for quantization-aware training and deployment using the Xilinx Vitis AI toolchain. The optimized model is successfully implemented on the Xilinx Kria KV260 platform and evaluated at $\times 2$ and $\times 4$ upscaling factors. Quantization-aware training minimizes accuracy degradation, with PSNR loss under one decibel and SSIM drop under ten percent. The final INT8 model achieves real-time inference exceeding 20 frames per second. Its memory usage is reduced from approximately 3.7 megabytes to just

0.3 megabytes, allowing it to run entirely within the FPGA’s on-chip memory without relying on external DRAM. These results demonstrate the practicality of deploying deep super-resolution models on edge AI platforms for efficient and scalable remote sensing applications.

ACKNOWLEDGMENT

This research was supported in part by the Multimedia Data Analytics and Processing Research Unit, Department of Electrical Engineering, Chulalongkorn University, Bangkok 10330, Thailand. It was also supported by the Graduate School, Chulalongkorn University, in commemoration of the auspicious occasion of Her Royal Highness Princess Maha Chakri Sirindhorn’s 60th birthday. Additionally, this work was funded by the Thailand Science Research and Innovation Fund, Chulalongkorn University (Grant Number: IND_FF_68_280_2100_039).

REFERENCES

- [1] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 2, pp. 295-307, 2015.
- [2] B. Lim, S. Son, H. Kim, S. Nah, and K. Mu Lee, "Enhanced deep residual networks for single image super-resolution," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2017, pp. 136-144.
- [3] A. Hajian and S. Aramvith, "AERU-Net: Adaptive Edge Recovery and Attention U-Shaped Network for Remote Sensing Image Super-Resolution," *IEEE Access*, 2025.
- [4] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 286-301.
- [5] S. Lei, Z. Shi, and W. Mo, "Transformer-based multistage enhancement for remote sensing image super-resolution," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1-11, 2021.
- [6] N. Sultan, A. Hajian, and S. Aramvith, "An advanced features extraction module for remote sensing image super-resolution," in *2024 21st International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, 2024: IEEE, pp. 1-6.
- [7] J. Wang, B. Wang, X. Wang, Y. Zhao, and T. Long, "Hybrid attention-based U-shaped network for remote sensing image super-resolution," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 61, pp. 1-15, 2023.
- [8] S. Santiwivat, A. Hajian, W. Ruangsang, and S. Aramvith, "Multiple Face Detection and Recognition for System-on-Chip FPGAs," in *2024 21st International Joint Conference on Computer Science and Software Engineering (JCSSE)*, 2024: IEEE, pp. 514-521.
- [9] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 3-19.
- [10] A. Dosovitskiy *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.
- [11] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in *Proceedings of the 18th SIGSPATIAL international conference on advances in geographic information systems*, 2010, pp. 270-279.
- [12] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600-612, 2004.