Scalable Inference for LArTPC Signal Processing with MobileU-Net and Overlap–Tile Chunking

Dikshant Sagar
Department of Computer Science
University of California, Irvine

Edgar Robles
Department of Computer Science
University of California, Irvine

Sergey Martynenko
Brookhaven National Laboratory

Yiwen Xiao
Department of Physics
University of California, Irvine

Jianming Bian
Department of Physics
University of California, Irvine

Pierre Baldi
Department of Computer Science
University of California, Irvine

Hokyeong Nam
Chung-Ang University

Jay Hyun Jo
Brookhaven National Laboratory

Wenqiang Gu
Brookhaven National Laboratory

Brett Viren
Brookhaven National Laboratory

Haiwang Yu
Brookhaven National Laboratory

Abstract

Liquid Argon Time Projection Chambers (LArTPCs) record ionization charge on wire planes over time, producing high-granularity wire-time images that undergo noise filtering and deconvolution to recover charge signals for downstream reconstruction [3]. Deep neural network region-of-interest (DNN-ROI) finding reframes ROI selection as semantic segmentation and has been shown to gate expensive processing effectively [14]. However, the reference U-Net-based DNN-ROI is memory-hungry and slow on CPUs, which remain the dominant resource for production processing in many HEP workflows. Prior work made inference feasible by rebinning the input by a factor of ten, reducing compute at the expense of spatio-temporal resolution [14]. We present a scalable DNN-ROI variant that retains full input resolution while substantially reducing memory and compute. First, we adopt a lightweight MobileNetV3 encoder within a U-Net decoder (*MobileU-Net*) to cut parameters and floating-point operations [6; 12; 8]. Second, we replace global downsampling with overlapping chunking (overlap-tile sliding-window inference with halo margins), enabling bounded-memory processing of large wire-time images without seam artifacts [11; 9]. To support reproducibility, the code used for training and experiments in this study has been released at `https://github.com/dikshantsagar/LArTPC-Segmentation`. On representative LArTPC data, the approach scales to larger inputs while largely maintaining ROI-finding performance relative to the original model. We detail methods and quantitative evaluations below.

Machine Learning and the Physical Sciences Workshop, NeurIPS 2025.

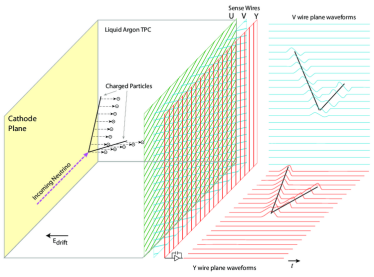

Figure 1: Schematic of neutrino event readouts from a LArTPC [3].

1 Introduction

LArTPCs provide fine-grained imaging by drifting ionization electrons to segmented wire planes, yielding waveforms that are filtered and deconvolved with field and electronics responses to estimate charge [3] (See Fig.1). Straightforward deconvolution can amplify low-frequency noise, so practical chains focus expensive processing on likely signal regions identified as ROIs [3].

Learning-based ROI selection has emerged as a robust and detector-agnostic alternative. In the DNN-ROI approach, wire-time data are assembled as images and a U-Net-style model predicts active regions that gate downstream processing [14]. While effective, the reference implementation exhibits high memory usage and nontrivial latency during CPU inference. Because many production environments in HEP still rely on CPU farms, prior work resorted to rebinning the input by an order of magnitude in time to make inference feasible [14]. This downscaling reduces compute but also degrades effective resolution, which can harm ROI fidelity and the quality of subsequent reconstruction.

This paper addresses the scalability bottleneck while preserving full resolution. We explore two complementary ideas:

1. **MobileU-NetV3 encoder.** Replace the heavy encoder with MobileNetV3 with inverted residual blocks and squeeze and excitation blocks, which provide accuracy-efficiency trade-offs when paired with lightweight decoders [6; 12; 8].
2. **Overlapping chunking.** Instead of downsampling, process large wire-time images with overlap-tile sliding-window inference. Each tile includes a halo region at least half the network receptive field; only the interior is retained, so stitched outputs are free of border artifacts while memory remains bounded [11; 9]. This can also be parallelized to further improve on time.

Contributions.

- A CPU-friendly DNN-ROI architecture that uses a MobileNetV3 encoder with a lightweight decoder to reduce parameters and FLOPs without sacrificing resolution and performance [6; 12; 8].
- A practical overlap-tile inference scheme with halos for LArTPC wire-time images that controls memory and avoids seam artifacts [11; 9].
- An empirical study on representative LArTPC data showing scalability to larger inputs with similar ROI-finding quality compared to the original model.

2 Methods

2.1 Approach overview

Figure 2 summarizes the pipeline. Input images are partitioned into chunks along the tick (time) dimension; each chunk is fed independently through the lightweight MobileU-Net. When overlapping

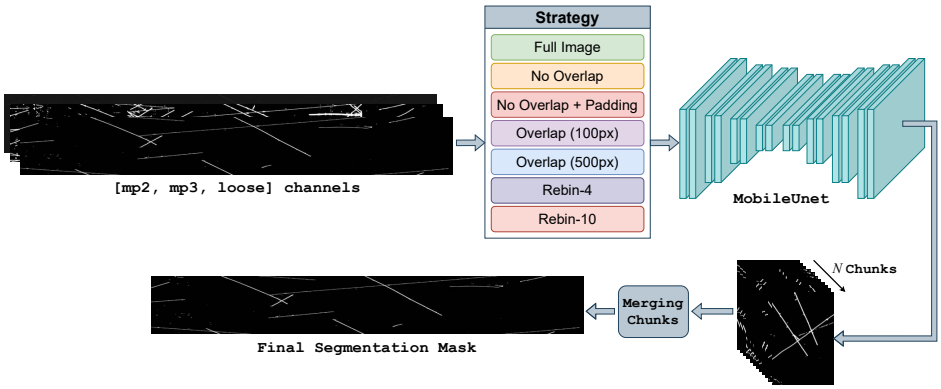

Figure 2: Schematic diagram of the proposed approach.

chunking is enabled, per-pixel scores in the overlap are averaged before stitching to form the final prediction.

Relative to the original DNN–ROI method [14], our approach (i) streams inputs via chunking to bound memory, optionally using overlaps to mitigate boundary artifacts, and (ii) replaces the U-Net encoder with a lightweight MobileNetV3 backbone (MobileU-Net) to improve computational efficiency.

Scope of this study. Our goal is to isolate *inference-time* optimizations. We therefore reuse network weights trained on full-resolution, non-rebinned inputs, and vary only the inference configuration (full image, non-overlapping chunks, overlapping chunks, or rebinning). This preserves a controlled comparison across strategies without confounding from additional training.

2.2 Dataset

The dataset is derived from raw LArTPC measurements of cosmic-ray events simulated with the Wire-Cell framework. It comprises approximately 8,000 events; for each event, the network receives four input images and one corresponding truth-label image. We use 80% of the data for training and the remainder as a held-out test set.

The simulation chain incorporates: Corsika [5] for cosmic-ray generation, Geant4 [4] for particle propagation, and LArSoft [13] as the event-processing framework. The detector configuration—active volume dimensions, number of channels, number of time ticks, and field-response parameters, follows the ProtoDUNE-HD design [10].

2.3 MobileU-Net

In LArTPC neutrino experiments, ROI segmentation is a critical preprocessing step for isolating candidate interaction areas from large, high-resolution detector images [2]. Traditional U-Net architectures combine multi-scale context with high-resolution detail, but their parameter counts and compute demands hinder deployment in real-time or resource-constrained scenarios [11]. MobileU-Net addresses this by using depthwise separable convolutions [7], reducing parameters and compute while preserving local and global spatial features.

2.4 Scalable and Efficient Segmentation for LArTPC Data

Full-resolution LArTPC images can reach 800×6000 pixels, making direct inference computationally expensive and memory-intensive. We evaluate several scalable strategies. First, *full-image inference* serves as a baseline, offering maximum spatial context at the largest memory footprint. Second, *non-overlapping chunk-based inference* divides each image into N horizontal segments processed independently, reducing memory substantially. To mitigate boundary artifacts, we also test *non-overlapping chunks with padding*, adding extra pixels around each segment to preserve edge context. Third, *overlapping chunk-based inference* allows adjacent chunks to share pixels (e.g., 100 or 500) to improve continuity at boundaries. Finally, we consider *rebinning*, downsampling the full image by a factor of 4 or 10 prior to inference, which lowers compute at the expense of fine detail. In addition,

Table 1: Comparison of U-Net and MobileU-Net across inference strategies on the LArTPC ROI task. Metrics: Intersection over Union (IoU), Purity, Efficiency on the test set; single-instance CPU memory and inference time. The definition of each strategy can be found in Sec 2.4.

		Test Set			**Single Instance**	
Model	**Strategy**	**IoU**	**Purity**	**Efficiency**	**Memory (MB)**	**Time (s)**
U-Net	Full image	0.83	0.89	0.88	6193.7	106.57
	No Overlap	0.82	0.89	0.87	936.2	100.47
	No Overlap + Padding	0.83	0.83	0.88	1028.5	126.96
	Overlap (100 px)	0.83	0.89	0.87	954.4	132.71
	Overlap (500 px)	0.83	0.89	0.87	954.5	552.28
	Rebin 4	0.55	0.52	0.83	1811.8	30.33
	Rebin 4 (Retrained)	0.79	0.83	0.88	1812.8	30.64
	Rebin 10	0.31	0.34	0.48	918.9	12.17
MobileU-Net	Full image	0.84	0.89	0.88	1719.7	14.41
	No Overlap	0.77	0.82	0.87	442.4	12.90
	No Overlap + Padding	0.75	0.85	0.84	463.5	12.85
	Overlap (100 px)	0.80	0.86	0.87	442.4	12.08
	Overlap (500 px)	0.82	0.88	0.88	442.4	50.28
	Rebin 4	0.63	0.44	0.85	635.0	4.29
	Rebin 4 (Retrained)	0.83	0.86	0.88	635.0	4.32
	Rebin 10	0.48	0.33	0.61	435.5	2.39

similar to the method paper [14], we trained two networks with rebinned images to study feature invariance under tick dimension scaling. The results for these two networks are labeled "Rebin 4 (Retrained)" in Table 1.

These chunking tactics offer multiple benefits. They allow deployment on hardware with limited resources, including systems without high-end GPUs. In particular, they enable feasible inference on CPUs, where processing a full-size image at once would be impractical due to memory and runtime constraints. After inference, segmented chunks are recombined to reconstruct the full-size ROI mask, ensuring continuity across boundaries.

To evaluate the efficiency of our approach, we benchmark both peak memory consumption and execution time using Fil Profiler [1], a Python-based memory analysis tool designed to provide detailed insights into memory allocation and performance bottlenecks.

3 Results

Table 1 summarizes segmentation performance and computational requirements for U-Net and MobileU-Net under different inference strategies. The main observations are as follows.

Architecture: Compared to U-Net, MobileU-Net achieves comparable or better accuracy with dramatically lower inference time, up to an order of magnitude faster for large inputs, while requiring substantially less memory (e.g., 1.7 GB vs. 6.2 GB for full-image inference). This efficiency arises from the use of inverted residual blocks with linear bottlenecks and embedded squeeze-and-excitation (SE) modules [6]. These blocks maintain representational power while reducing the number of high-dimensional feature maps and expensive convolution operations. The SE modules further enhance channel-wise feature recalibration with minimal computational overhead. Together, these design choices yield a model that is both compact and computationally efficient, offering a more favorable trade-off between accuracy and inference cost, particularly for high-resolution inputs.

Chunking: For both models, full-image inference yields the highest or near-highest IoU; however, MobileU-Net achieves this with substantially lower memory usage (about 1.7 GB) compared to U-Net (over 6 GB). Non-overlapping chunking further reduces memory consumption by roughly an order of magnitude for each model with some accuracy loss, indicating clear boundary artifacts. Overlapping chunking overcomes boundary effects and preserves near full-image accuracy but increases inference

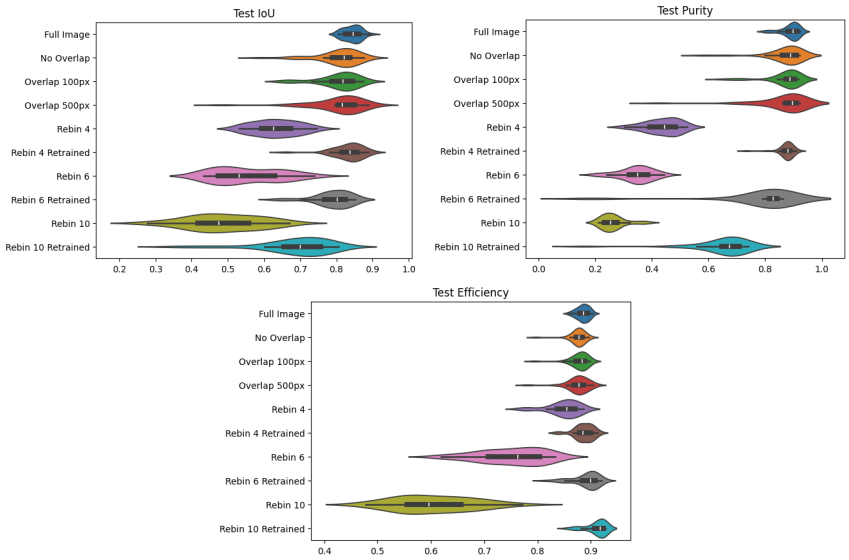

Figure 3: MobileUnet's IoU, Purity, and Efficiency distribution violin plots over the test set via multiple strategies.

time considerably as the overlap size grows from 100px to 500px, due to redundant computations across patch boundaries.

Rebinning: Rebinning prior to inference provided the greatest reductions in speed and memory cost, but at the expense of significant accuracy degradation. Comparing "Rebin 4 (Retrained)" with "Rebin 4," the computational cost remains unchanged, as expected, while performance improves remarkably. This demonstrates that the learned features are not scale-invariant, necessitating retraining when applying different scaling.

Metrics Uncertainity: Figure 3 illustrates the distribution of test metrics across runs for MobileU-Net under different inference strategies. The violin plots show that performance variability is minimal for full-image and chunked inference, indicating stable predictions across test samples. In contrast, rebinning introduces greater uncertainty, especially for unretrained models, reflecting sensitivity to input resolution changes. Retraining on rebinned data consistently narrows these distributions, confirming improved robustness and scale adaptation. Overall, the low variance in IoU, purity, and efficiency for non-rebinned and retrained configurations demonstrates that MobileU-Net produces reliable, consistent outputs across diverse inference settings.

4 Conclusion

We studied the trade-off between segmentation performance and computational efficiency for CPU-dominated facilities. Two full-resolution models, a standard U-Net and a MobileU-Net, were evaluated under inference configurations involving chunking and rebinning.

Findings. (1) MobileU-Net achieves comparable or better accuracy while requiring substantially less memory and providing up to an order of magnitude faster inference than U-Net; (2) Non-overlapping chunking greatly reduces memory and compute cost with only some accuracy loss; (3) overlapping chunking mitigates boundary artifacts but increases runtime as overlap size grows; (4) models trained on 'un'-rebinned data generalize poorly to rebinned inputs, confirming that learned features are not scale-invariant.

Takeaway. MobileU-Net with overlapping (100px) chunking offers the best balance between accuracy, speed, and memory efficiency, making it a practical choice for large-scale deployment on CPU-based production systems, especially where chunk-based inference can be parallelized, further reducing inference time. Alternatively, using a rebin factor of 4 achieves nearly identical accuracy to the full image inference while providing substantially faster CPU performance.

References

[1] Fil: A memory profiler for Python.

[2] ABRATENKO, P., ALRASHED, M., AN, R., ANTHONY, J., ASAADI, J., ASHKENAZI, A., BALASUBRAMANIAN, S., BALLER, B., BARNES, C., BARR, G., ET AL. Semantic segmentation with a sparse convolutional neural network for event reconstruction in microboone. *Physical Review D 103*, 5 (2021), 052012.

[3] ADAMS, C., ET AL. Ionization electron signal processing in single phase LArTPCs. Part I. Algorithm Description and quantitative evaluation with MicroBooNE simulation. *JINST 13*, 07 (2018), P07006.

[4] AGOSTINELLI, S., ET AL. GEANT4 - A Simulation Toolkit. *Nucl. Instrum. Meth. A 506* (2003), 250–303.

[5] ENGEL, R., HECK, D., HUEGE, T., PIEROG, T., REININGHAUS, M., RIEHN, F., ULRICH, R., UNGER, M., AND VEBERIČ, D. Towards a Next Generation of CORSIKA: A Framework for the Simulation of Particle Cascades in Astroparticle Physics. *Comput. Softw. Big Sci. 3*, 1 (2019), 2.

[6] HOWARD, A., SANDLER, M., CHU, G., CHEN, L.-C., CHEN, B., TAN, M., WANG, W., ZHU, Y., PANG, R., VASUDEVAN, V., ET AL. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF international conference on computer vision* (2019), pp. 1314–1324.

[7] HOWARD, A. G., ZHU, M., CHEN, B., KALENICHENKO, D., WANG, W., WEYAND, T., ANDREETTO, M., AND ADAM, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861* (2017).

[8] LAIBACHER, T., WEBER, T., AND WILHELM, M. H. F. M2U-Net: Effective and Efficient Retinal Vessel Segmentation for Real-World Applications. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (2019), pp. 115–123.

[9] MAJURSKI, M., MANESCU, A., SCHMIDT, P. R., AND FOX, M. H. Efficient Tile-Based Image Segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)* (2021), pp. 663–672.

[10] MANZANILLAS VELEZ, L., ET AL. Status of protodune-ii. *Proceedings of the 42nd International Conference on High Energy Physics (ICHEP2024)* (2025). Includes description of ProtoDUNE-HD horizontal-drift design.

[11] RONNEBERGER, O., FISCHER, P., AND BROX, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)* (2015), pp. 234–241.

[12] SANDLER, M., HOWARD, A., ZHU, M., ZHMOGINOV, A., AND CHEN, L.-C. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2018), pp. 4510–4520.

[13] SNIDER, E. L., AND PETRILLO, G. LArSoft: Toolkit for Simulation, Reconstruction and Analysis of Liquid Argon TPC Neutrino Detectors. *J. Phys. Conf. Ser. 898*, 4 (2017), 042057.

[14] YU, H., ET AL. Augmented signal processing in Liquid Argon Time Projection Chambers with a deep neural network. *JINST 16*, 01 (2021), P01036.