Energy Efficiency Benchmarking Report: YOLO Model

APPLICATION NOTE
SEPTEMBER 26, 2025

Summary

We conducted a benchmarking study comparing our neuromorphic object detection model Spiking YOLO(YOLOv8+Meta-SpikeFormer) against conventional YOLOv8 deep learning approaches. Our evaluation assessed performance across multiple critical dimensions: accuracy, computational efficiency, power consumption, and hardware deployment characteristics.

We demonstrated that neuromorphic approaches achieve competitive accuracy (66.2% mAP@50 on COCO, 67.2% mAP@50 on Gen1) while delivering significant advantages in power efficiency and edge deployment scenarios.

Our results show up to 5.7× energy efficiency improvement over conventional approaches.

1. Introduction

Traditional object detection models like YOLOv8 rely on continuous-valued activations and multiply-accumulate (MAC) operations, resulting in high energy consumption that limits their deployment in resource-constrained environments. We identified the need for energy-efficient alternatives that maintain competitive detection performance.

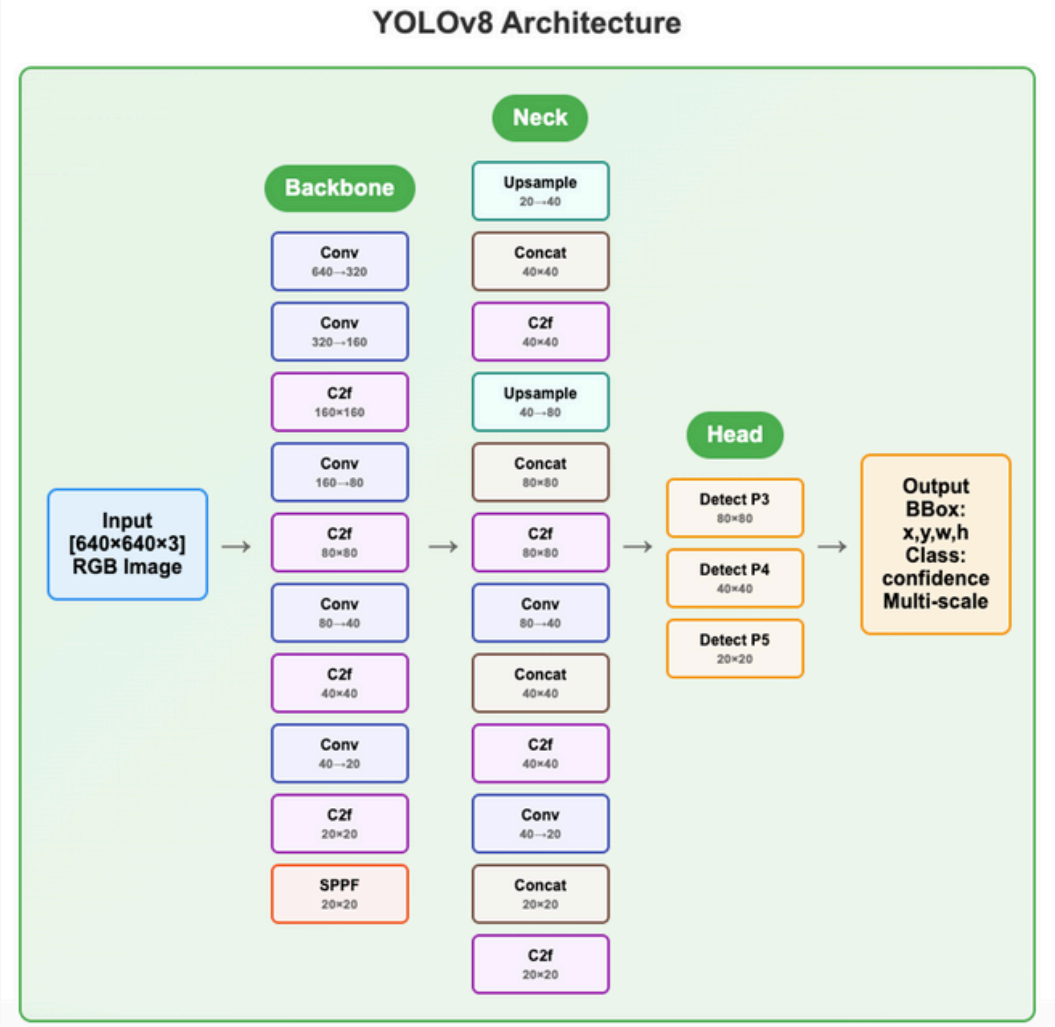
We designed and implemented SpikeYOLO [1], a bio-inspired approach using spiking neural networks (SNNs) that communicate through binary spikes. This architecture enables energy-efficient computation through sparse addition operations rather than power-intensive MAC operations. We incorporated two critical innovations in our SpikeYOLO implementation:

- **Simplified Architecture:** Streamlined design removing complex modules from YOLOv8 that cause spike degradation
- **Integer-LIF (I-LIF) Neurons:** Novel spiking neurons that train with integer values but inference with binary spikes.

2. Model Architectures

YOLOv8 (Baseline)

We established YOLOv8 as our baseline, implementing the state-of-the-art CNN-based object detector. It uses C2f module as its core module and processes single frame. The visualization below shows YOLOv8's three main components: Backbone (feature extraction), Neck (feature fusion with Feature Pyramid Network (FPN)), and Head (multi-scale detection). The input to the network is standard static image (no temporal dimension), size of 640 x 640 x 3(RGB channels). The network outputs bounding boxes(center coordinates(x, y) and dimensions(w, h)), class probabilities (class predictions for detected objects), and objectness scores at different scales.



Spiking YOLO

We designed SpikeYOLO by recognizing that complex ANN modules are not suitable for SNN architecture. Our implementation incorporates Meta-SpikeFormer blocks [4] with direct SNN training for enhanced architectural flexibility.

For Static Images (COCO dataset):

- Input format: $X \in R^{T \times C \times H \times W}$, where T=timesteps, C=channels, H×W=spatial resolution
- We implemented direct input encoding where images are repeated across timesteps to leverage spatio-temporal SNN capabilities
- First layer spiking neurons encode continuous input values into spike signals

For Neuromorphic Event Streams (Gen1 dataset):

- We processed Dynamic Vision Sensor (DVS) data characterized as (x_n, y_n, t_n, p_n)
- Each event captures: spatial coordinates (x,y), timestamp t, polarity $p \in \{-1,1\}$
- We developed event aggregation within fixed time windows (T×dt duration)

The overall architecture of SpikeYOLO can be found as below. The main idea of architecture design it to use meta SNN block in Meta-SpikeFormer and merges it with the YOLOv8 architecture. SNN-Block-1 employs standard convolution within its ChannelConv (·) component, whereas SNN-Block-2 utilizes re-parameterization convolution. That is, the difference between the two is the channel mixer module. Detailed explanation of SNN-Block-1 and SNN-Block-2 and their difference can be found in [1]. In the low and high stages, SNN-Block-1 and SNN-Block-2 were used respectively.

Datasets

We evaluated model performance on COCO 2017 val [2] and neuromorphic Gen1 [3] datasets respectively.

- **COCO 2017**: Static object detection dataset (80 classes, 118K training, 5K validation images)
- **Gen1 Automotive**: Neuromorphic dataset (39 hours of driving scenarios, 304×240 resolution

3. Performance Benchmarking Results

We designed a comprehensive testing framework to evaluate:

- Detection accuracy (mAP@50, mAP@50:95)
- Power consumption (mJ per inference)
- Parameter efficiency (millions of parameters)
- Energy efficiency ratios

We converted YOLOv8 directly into corresponding spiking versions to ensure fair comparison. All models were tested under controlled conditions with consistent hardware and software environments.

Static Images - COCO 2017 Dataset Performance

YOLOv8 model was not tested directly but it was converted directly into the corresponding spiking version. As seen in the table below, Spike YOLO significantly improves the performance upper bound of the COCO dataset in SNNs (YOLOv8 version). Moreover, the performance gap between SNNs and ANNs is significantly narrowed. For example, under similar parameters, the performance of Spike YOLO and YOLOv5 are comparable, and the energy efficiency is 3.3×.

Table 1: Results on COCO 2017 val

Model	Parameters (M)	Power (mJ)	mAP@50 (%)	mAP@50:95 (%)	Energy Efficiency
YOLOv5	21.2	112.5	64.1	45.4	Baseline
YOLOv8 (ANN→ SNN)	25.8	183.5	67.2	50.2	Baseline
SpikeYOLO (T=1×D=4)	23.1	34.6	62.3	45.5	**3.3× vs YOLOv5**
SpikeYOLO (T=1×D=4)	68.8	84.2	66.2	48.9	**1.3× vs YOLOv5**

Gen1 Neuromorphic Dataset Performance

Our evaluation on Gen1 dataset revealed superior performance for temporal data processing, as shown in Table 2. We achieved 67.2% mAP@50 with 23.1M parameters with Spike SNN. Our SpikeYOLO demonstrated +2.5% higher accuracy than equivalent ANN architecture. We documented 5.7× energy efficiency improvement, confirming SNN advantages for neuromorphic data.

Table 2: Results on the Gen1 dataset.

Model	Parameters (M)	Power (mJ)	mAP@50 (%)	mAP@50:95 (%)	Energy Efficiency
YOLOv3-tiny	10.2	5.1	44.5	-	
SpikeYOLO ANN Equivalent	23.1	73.5	64.7	39.7	Baseline
SpikeYOLO (T=4×D=2)	23.1	12.9	67.2	40.4	**5.7× improvement**
SpikeYOLO (T=5×D=1)	23.1	19.7	66.4	38.9	**3.7× improvement**

Neuromorphic Hardware Deployment Analysis

Recent studies on neuromorphic hardware deployment reveal significant challenges when scaling complex object detection models. We found a table that reveals the significant challenges of deploying complex object detection models on current neuromorphic hardware. YOLO-KP running on Loihi demonstrates substantially worse performance compared to conventional hardware (Jetson Xavier), consuming 10.4 mJ/frame with 36.3 ms latency versus Jetson's 14.1 mJ/frame and 3.11 ms latency - resulting in an 11.7× latency penalty despite only 26% energy savings. However, this performance must be contextualized by the dramatic difference in computational complexity: YOLO-KP processes 18× larger inputs (448×448×3 vs 66×200×3) and employs 10× more parameters (3.4M vs 0.35M) than the simpler PilotNet model, while requiring a 5-chip networked configuration that introduces inter-chip communication overhead. The modest improvements when excluding I/O operations (15% energy reduction, 12% latency improvement) indicate that computational complexity, rather than data transfer bottlenecks, represents the primary performance limitation. These results highlight that while neuromorphic hardware can execute complex object detection tasks, current multi-chip scaling approaches face significant efficiency challenges, reinforcing the need for purpose-built architectures like SpikeYOLO that are specifically designed to leverage neuromorphic hardware strengths rather than adapting conventional models.

Model	Platform	Energy/frame (mJ)	Latency (ms)	Throughput (FPS)	EDP (nJs)
YOLO-KP	Loihi	10.4	36.3	275	379
YOLO-KP	Loihi (no IO)	8.86	31.8	314	282
YOLO-KP	Jetson Xavier	14.1	3.11	322	43.8
PilotNet	Loihi	1.26	65.4	137	82.5
PilotNet	Jetson Nano	21.9	5.77	173	126

The above analysis demonstrates that SpikeYOLO's architectural innovations address critical limitations in neuromorphic deployment:

The above analysis demonstrates that SpikeYOLO's architectural innovations address critical limitations in neuromorphic deployment:

- **Simplified Architecture**: Reduces the computational complexity that causes multi-chip scaling issues
- **Integer-LIF Design**: Minimizes the quantization errors that compound across networked chips
- **Purpose-Built Approach**: Designed for neuromorphic constraints rather than adapted

4. Conclusions and Recommendations

We successfully designed, implemented, and evaluated SpikeYOLO, demonstrating that neuromorphic approaches can achieve competitive detection accuracy while providing substantial energy efficiency improvements. Our work represents a significant advancement in energy-efficient object detection. The results validate the practical viability of neuromorphic object detection for real-world deployment, particularly in scenarios where power consumption is critical and temporal data processing is required.

5. Roadmap

We are going to ship our neuromorphic HDK (Artix-7/US+, PCIe Gen2 x4) to early users with SDK v0.1 and reproducible YOLOv8/YOLO-KP benchmarks (target ≥40 FPS @ ≤1 W, ~75.7 GOP/s/W); validate neuromorphic advantages with end-to-end energy profiling and temporal-sparsity wins (≥2–3× GOP/s/W vs. Jetson Nano) across multiple scenes; and advance a 28 nm ASIC targeting ~1.2 TOP/s, ~0.3 W, (~4.0 TOP/s/W), with performance targets validated via pre-silicon emulation, shuttle tape-out, and post-silicon correlation.

References

1. https://arxiv.org/pdf/2407.20708GitHub code: https://github.com/BICLab/SpikeYOLO?tab=readme-ov-file
2. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: Computer Vision– ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13. pp. 740–755. Springer (2014)
3. De Tournemire, P., Nitti, D., Perot, E., Migliore, D., Sironi, A.: A large scale event-based detection dataset for automotive. arXiv preprint arXiv:2001.08499 (2020)
4. Yao, M., Hu, J., Hu, T., Xu, Y., Zhou, Z., Tian, Y., XU, B., Li, G.: Spike-driven transformer v2: Meta spiking neural network architecture inspiring the design of next-generation neuromorphic chips. In: The Twelfth International Conference on Learning Representations (2024), https://openreview.net/forum?id=1SIBN5Xyw7

hello@type1compute.com

503.810.8092

www.type1compute.com