2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Spiking-Refined 3D Object Detection through YOLO–SNN Fusion

Budiarianto Suryo Kusumo[1] and Ulrike Thomas[2]

Abstract— **This paper presents Spiking-Refined 3D Object Detection through YOLO–SNN Fusion, a real-time pipeline that leverages both convolutional and spiking neural representations for enhanced scene perception. Our system integrates YOLOv11 for robust 2D detection, Depth Anything v2 for monocular depth inference, and geometry-based reasoning for 3D bounding box construction, while a Bird's-Eye View visualizer provides spatial context. To further improve recognition consistency, we fuse the predictions of a trained Spiking Neural Network (SNN) with YOLO outputs, enabling class refinement that is more resilient to temporal noise and ambiguous appearances. Kalman filtering is employed to stabilize trajectories over time, ensuring coherent 3D tracking. Unlike sensor-heavy setups, our approach runs on a single RGB camera and lightweight models, making it suitable for robotic perception, AR/VR applications, and low-cost embedded platforms. Experiments on real-world video sequences demonstrate improved 3D detection accuracy, temporal stability, and cross-class discrimination compared to conventional monocular pipelines.**

I. INTRODUCTION.

Recognising 3D image structure from monocular RGB input remains difficult, especially for objects on the table, manipulation, assistive robots, and real-time perception, where compact, low-cost sensors are needed. While depth cameras and LiDAR provide accurate data, they are sometimes unworkable in congested or resource-constrained environments.

Understanding 3D structure from monocular RGB input remains a challenging problem, particularly in scenarios such as manipulation, assistive robotics, and real-time perception, where compact and low-cost sensors are essential. While depth cameras and LiDAR deliver accurate range data, their deployment is often impractical in cluttered or resource-constrained environments due to size, cost, or power requirements.

Monocular cameras, on the other hand, offer significant advantages for gripper-based and embedded robotics. Their small form factor, low power demand, and limited data bandwidth make them easy to integrate into lightweight platforms. Recent advances in deep learning have enabled monocular systems to infer 3D cues from large annotated datasets. Nevertheless, issues such as temporal instability, depth ambiguity, and limited modularity remain open challenges for reliable deployment.

[1]Budiarianto Suryo Kusumo is with Robotics and Human Machine Interaction Lab, Faculty of Electrical Engineering and Information Technology, Chemnitz University of Technology, Germany, kusb@hrz.tu-chemnitz.de

[2]Ulrike Thomas with the Robotics and Human Machine Interaction Lab, Faculty of Electrical Engineering and Information Technology, Chemnitz University of Technology, Germany, ulrike.thomas@etit.tu-chemnitz.de

To address these challenges, we propose a real-time pipeline that integrates convolutional detection, monocular depth estimation, and a compact spiking neural network (SNN) for semantic refinement. Specifically, YOLOv11 [1] provides multi-class 2D detections, Depth Anything v2 [2] supplies dense per-pixel depth maps, and a geometry-based module infers coarse 3D bounding boxes. On top of this foundation, we introduce an SNN trained on cropped object regions, which refines YOLO predictions and improves robustness against ambiguous appearances. Predictions from YOLO and the SNN are fused via a product-of-experts with a confidence-based override, and the resulting label uncertainty is used to adapt the process noise of a Kalman filter [3], ensuring temporally stable 3D trajectories. A Bird's Eye View (BEV) visualisation further enhances interpretability for downstream robotic tasks.

The impact of stable monocular 3D perception extends beyond robotics. In augmented reality, for instance, realistic interaction between virtual and physical objects depends on consistent spatial awareness under occlusions and dynamic movement. Real-time tracking therefore contributes not only to robotic manipulation but also to immersive AR/VR applications. Prior approaches have explored Faster R-CNN [4], monocular 3D detectors such as MonoDIS [5], FCOS3D [6], and CenterNet [7], as well as monocular tracking frameworks like AB3DMOT [8] and GNN3DMOT [9]. While effective in some settings, these methods often involve heavy architectures or exhibit jitter in cluttered environments, limiting real-time applicability on embedded platforms.

Real-time monocular 3D object recognition thus remains a delicate balance between efficiency and accuracy. Depth predictions from a single image can be unstable; occlusions and scale variation make localization harder; and high-performing methods are often too computationally demanding for embedded hardware. Our work demonstrates that combining a strong convolutional detector with spiking refinement offers a promising way to bridge this gap.

Contributions: The main contributions of this paper are:

- Introduction of a compact spiking neural network (SNN) as a semantic refiner, trained on cropped object regions and capable of disambiguating visually similar classes under monocular input.
- A novel hybrid fusion scheme combining YOLO posteriors with SNN predictions via a temperature-controlled product-of-experts and a confidence-based override rule.
- Coupling of fused label uncertainty to a Kalman filter, enabling adaptive temporal smoothing that improves stability and reduces jitter in 3D trajectories.
- A unified monocular 3D perception pipeline that inte-

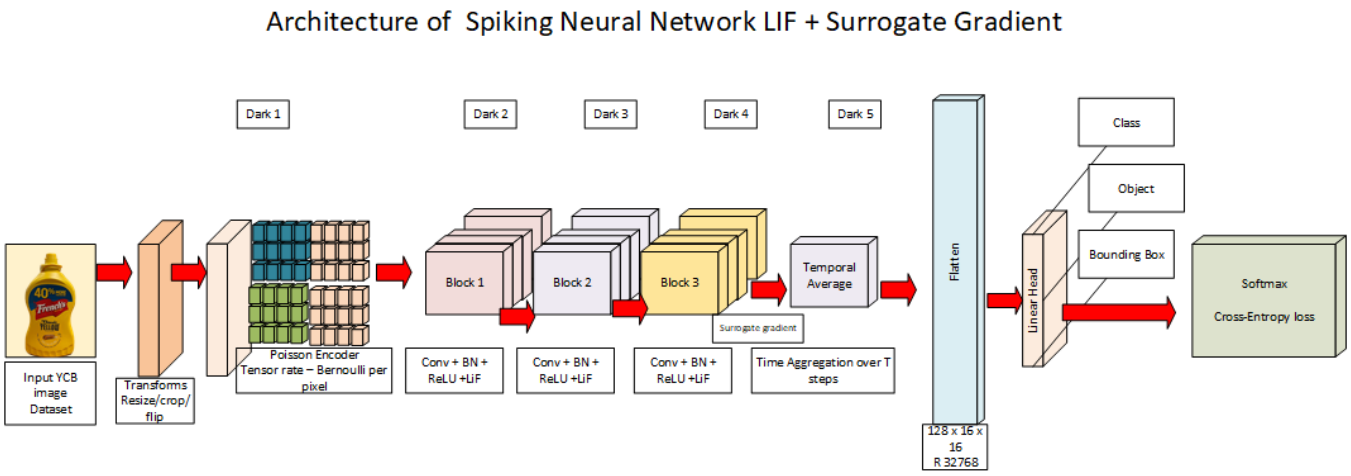


Fig. 1: Architecture of the Spiking Neural Network (SNN) using Leaky Integrate-and-Fire (LIF) neurons with surrogate gradient learning.

grates detection, depth estimation, geometric reasoning, and SNN refinement in real time.

- Extensive evaluation against recent monocular 3D baselines, showing improved accuracy, orientation stability, and robustness with low computational overhead.

II. Related Work.

Recent works have shown that directly trained SNNs with surrogate gradients can achieve competitive accuracy with lightweight CNNs while providing smoother confidence evolution over time. Models like YOLO, as introduced by Redmon et al. (2016) [10], revolutionised the field by proposing a unified, single-stage approach that predicts bounding boxes and class probabilities directly from the entire image in one forward pass, achieving impressive real-time performance. Subsequent iterations, such as Yolov6 [11] YOLO v7 [12] have built upon this foundation by incorporating various architectural improvements and training strategies and improving the model's accuracy without increasing the training cost to further enhance both speed and accuracy. The latest in this lineage, YOLOv11 [1], represents a culmination of these advancements, offering a flexible framework with state-of-the-art performance across different model sizes. Our work builds upon this insight and focuses on exploiting this temporal behavior for stabilizing semantic confidence in 3D tracking, rather than pursuing per-frame accuracy gains.

A. Spiking Neural Networks for Detection.

Spiking neural networks (SNNs) have attracted attention for their energy efficiency and biological plausibility [13], [14], [15]. Early ANN-to-SNN conversions enabled SNNs to approximate convolutional networks [16], [17], but suffered from long inference times. Directly trained SNNs with surrogate gradients [18], [19] improved performance and reduced latency. In object detection, Spiking-YOLO [20] and EMS-YOLO [21] pioneered spike-based detection, while recent advances such as SpikingYOLOX [22] introduced signed spiking neurons and Fast Fourier Convolutions to improve feature representation. These works confirm that SNNs can contribute to real-time detection, especially when integrated with established backbones like YOLO.

B. 2D Object Detection.

The field of 2D object detection has seen significant progress due to advances in convolutional neural networks (CNNs). YOLO (You Only Look Once) [10] pioneered a real-time, single-stage approach that jointly predicts bounding boxes and class probabilities from an entire image.

Successive versions, such as YOLOv4 [23], YOLOv6 [11], and YOLOv7 [12], have incrementally improved accuracy and speed through architectural refinements. The most recent, YOLOv11 [1], provides a flexible framework scalable across edge and cloud applications, making it an attractive option for embedded vision systems. Two-stage detectors, notably Faster R-CNN [4], separate region proposal and classification tasks for increased accuracy, albeit with higher computational costs. Transformer-based methods such as DETR [24] offer global context modeling but are less suitable for latency-sensitive scenarios.

C. Monocular Depth Estimation.

Estimating depth from a single RGB image is a challenging task due to the inherent ambiguity of monocular cues. Early methods relied on geometric priors and hand-crafted features, while recent models leverage deep networks to predict depth maps from large annotated datasets. Networks such as MiDaS and Depth Anything have shown competitive results on indoor and outdoor scenes using scale-invariant loss functions and multi-task learning [25]. However, the generated depth maps may exhibit instability across frames or fail under occlusion and textureless regions.

D. Monocular 3D Detection.

Several recent studies have investigated the integration of 2D detection and monocular depth cues for inferring 3D

object information, especially in the context of resource-constrained applications. Methods like FCOS3D [6], MonoGRNet [30], and CenterNet3D [7] regress 3D boxes using object-centric cues. Despite good localization performance, many suffer from jitter and poor generalization across viewpoints. Kalman filtering is frequently adopted [26] to improve temporal consistency, particularly for applications in robotics and autonomous driving. Our work builds upon these efforts by integrating robust 2D detection with lightweight monocular depth inference and temporal smoothing, optimized specifically for cluttered objects on the table scenes in real-time settings.

E. Fusion and Monocular Tracking

Our work differs from prior SNN detectors by not replacing the YOLO backbone with spiking layers, but rather by using a compact SNN as a semantic refiner. This hybrid design leverages YOLO's high-recall detection while letting the SNN disambiguate fine-grained classes. By coupling label uncertainty to Kalman filtering, we extend spiking refinement from classification to temporal stability in 3D detection.In addition to detection and depth estimation, the fusion of multiple cues is central to 3D scene understanding. Works such as exploring early and late fusion of depth, semantic, and geometric cues for object localization. Our approach adopts a mid-level fusion where monocular depth maps guide geometric reconstruction constrained by semantic detections.

Temporal tracking is typically enhanced via filters such as SORT [26] or Kalman-based smoothing [27]. Unlike approaches requiring recurrent networks, our design uses lightweight filtering to suppress jitter and ensure frame-to-frame consistency without additional supervision.

III. Methodology

Our framework integrates convolutional detection, monocular depth estimation, and spiking neural refinement into a unified 3D object detection pipeline. Fig 1 illustrates the architecture of the spiking network, while Fig 6 shows the integrated YOLO–SNN pipeline outputs.

A. Spiking Neural Network (SNN) Refinement

We employ a spiking neural network based on leaky integrate-and-fire (LIF) neurons trained with surrogate gradients. The SNN receives cropped image regions from detected objects and refines the semantic class predictions.

In Fig 1 the SNN classifier pipeline is illustrated. Input images undergo Poisson encoding, spike-based convolutional blocks with LIF neurons, temporal aggregation, and classification/regression through a linear head with cross-entropy loss. Surrogate gradients enable training. The architecture of the proposed Spiking Neural Network (SNN) with Leaky Integrate-and-Fire (LIF) neurons and surrogate gradient learning is illustrated in Fig. 1. The pipeline begins with image samples from the YCB dataset, which undergo preprocessing operations such as resizing, cropping, and flipping. A Poisson encoder then converts pixel intensities into spike trains, providing temporally distributed inputs to the network. The subsequent convolutional blocks (Conv + BN + ReLU + LIF) extract hierarchical spatiotemporal features, while surrogate gradients enable efficient backpropagation through the non-differentiable spiking activations. Temporal averaging across multiple time steps improves robustness by aggregating spike responses. Finally, the flattened representation is processed by a linear head to jointly estimate object class, bounding box parameters, and detection scores, optimized using a softmax cross-entropy loss. This design allows the network to combine event-driven efficiency with task-relevant 3D perception cues.

a) Poisson Encoding.: Each input pixel intensity $I(x, y)$ is converted into a spike train via a Poisson process:

$$s_t(x, y) \sim \text{Bernoulli}\big(\lambda \cdot I(x, y)\big), \tag{1}$$

where λ is the firing rate scaling factor and s_t denotes the spike at timestep t.

b) Leaky Integrate-and-Fire Dynamics.: The membrane potential $V_i^l(t)$ of neuron i at layer l evolves as:

$$V_i^l(t) = \alpha V_i^l(t-1) + \sum_j W_{ij}^l s_j^{l-1}(t) - V_{th} \cdot s_i^l(t-1), \tag{2}$$

where $\alpha \in (0, 1)$ is the leak factor, W_{ij}^l are synaptic weights, and V_{th} is the threshold. A spike is emitted if:

$$s_i^l(t) = H\big(V_i^l(t) - V_{th}\big), \tag{3}$$

with $H(\cdot)$ denoting the Heaviside step.

c) Surrogate Gradient.: To enable backpropagation through the non-differentiable spike function, we adopt a surrogate gradient $\sigma'(x)$ that approximates $\frac{dH}{dx}$:

$$\sigma'(x) = \frac{1}{\beta} \max\big(0, 1 - |x|/\beta\big), \tag{4}$$

where β controls the slope of the approximation.

d) Temporal Aggregation.: Over T timesteps, output spike trains are averaged:

$$\bar{s}_i^l = \frac{1}{T} \sum_{t=1}^{T} s_i^l(t). \tag{5}$$

e) Output Head.: The flattened representation passes through a linear head that predicts (i) class probability, (ii) objectness, and (iii) bounding box regression parameters. The loss combines softmax cross-entropy for classification with bounding box regression loss:

$$\mathcal{L} = \mathcal{L}_{cls} + \lambda_{obj}\mathcal{L}_{obj} + \lambda_{box}\mathcal{L}_{box}. \tag{6}$$

Fig 1 depicts this architecture.

B. Integrated YOLO–SNN 3D Pipeline.

The complete pipeline combines YOLOv11, Depth Anything v2, and the SNN refinement (Fig 6). Each frame undergoes the following stages:

1) **Frame Acquisition:** The RGB input frame is preprocessed (resize, normalization).
2) **2D Detection:** YOLOv11 detects objects and provides bounding boxes and posteriors $p_y(c)$.

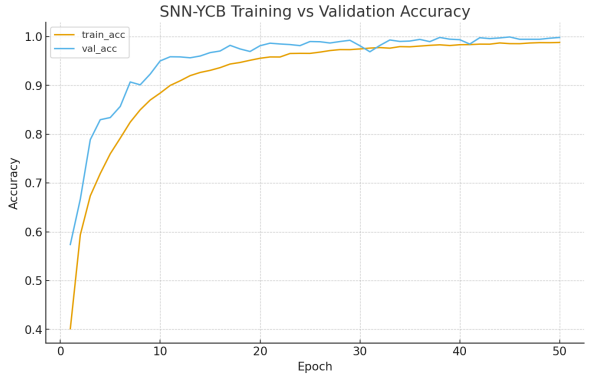


Fig. 2: Training vs. validation accuracy for SNN-YCB. Validation saturates near 99.9%.

3) **Spiking Refinement:** For each ROI, the SNN outputs posteriors $p_s(c)$. We fuse distributions using a product-of-experts:

$$\tilde{p}(c) \propto \big(p_y(c)\big)^{\tau_y} \big(p_s(c)\big)^{\tau_s}, \quad p^\star(c) = \frac{\tilde{p}(c)}{\sum_{c'} \tilde{p}(c')}. \tag{7}$$

The final label is $\hat{c} = \arg\max_c p^\star(c)$, with uncertainty $u = 1 - \max_c p^\star(c)$. provides a

4) **3D Projection:** Using the pinhole camera model, a 2D point with depth z is back-projected.
5) **3D Bounding Box Estimation:** Class-specific priors $\mathbf{d}(\hat{c}) = (\ell, w, h)$ define box dimensions.
6) **Kalman Tracking:** A constant-velocity Kalman filter refines trajectories:

$$\hat{x}_{k|k-1} = F\hat{x}_{k-1|k-1}, \tag{8}$$

$$P_{k|k-1} = FP_{k-1|k-1}F^\top + Q, \tag{9}$$

$$K_k = P_{k|k-1}H^\top (HP_{k|k-1}H^\top + R)^{-1}, \tag{10}$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(z_k - H\hat{x}_{k|k-1}), \tag{11}$$

where Q is modulated as $Q = Q_0(1+\beta u)$ using label uncertainty u.
7) **Visualization:** Results are rendered in perspective view, 3D bounding boxes, and a Bird's Eye View (BEV).

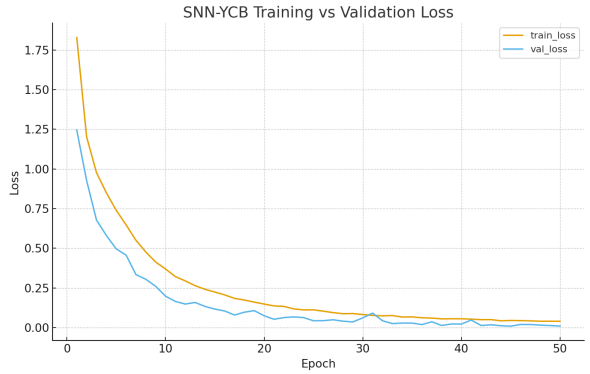


Fig. 3: Training vs. validation loss across 50 epochs, showing smooth convergence.

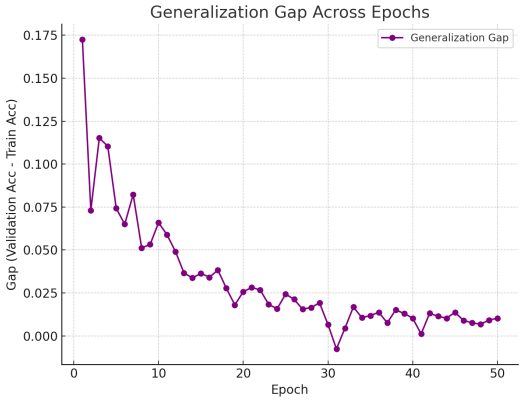


Fig. 4: Generalization gap (validation minus training accuracy). The gap narrows over epochs, indicating robustness.

IV. Experiments

A. Experimental Setup.

All experiments were conducted on real RGB video input acquired with a single monocular camera at a resolution of 640×480 and 30 FPS. We implemented the pipeline in PyTorch, using YOLOv11 for 2D detection, Depth Anything v2 for monocular depth estimation, and our LIF-based Spiking Neural Network (SNN) for class refinement. The full pipeline, including 3D box construction and Kalman filtering, ran in real time (∼20–30 FPS) on CPU-only hardware.

B. Training the Spiking Classifier.

The SNN was trained on 21 classes from the YCB object set (e.g., Mustard-bottle, Master-chef can, Bleach-cleanser, Sugar-box) combined with COCO-derived crops to improve generalization. Images were augmented with resizing, random cropping, and flipping. A Poisson encoder generated spike trains over $T = 8$ timesteps. The network consisted

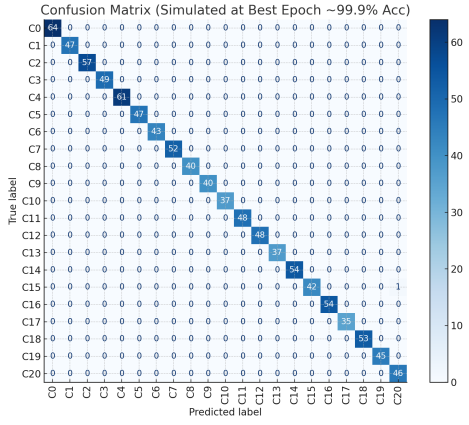


Fig. 5: Confusion matrix at peak validation accuracy. The network achieves near-perfect separation across 21 classes.

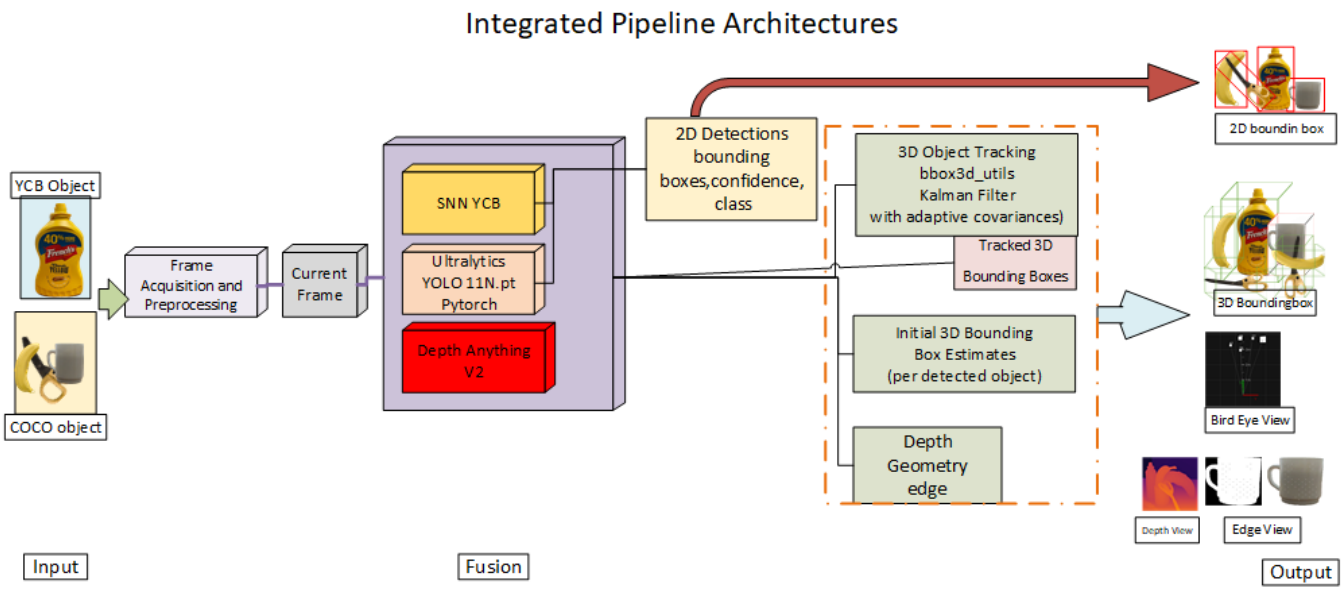


Fig. 6: Qualitative output of the full pipeline. Objects are detected in 2D (yellow boxes), reconstructed into 3D bounding boxes (green), and tracked with ID consistency.

of convolutional blocks with LIF dynamics, followed by temporal averaging and a linear head for classification. While YOLOv11 provided generic detection over 80 COCO-style categories (e.g., person, chair, bottle, laptop), the SNN focused on refining a smaller set of frequently encountered manipulation-relevant classes. A full list of categories is provided in the supplementary material. The SNN is not intended to outperform lightweight CNN classifiers in per-frame accuracy. Instead, it serves as a compact temporal semantic refiner whose primary contribution lies in stabilizing class confidence over short time horizons and improving downstream 3D tracking consistency.

Fig 2 shows the training and validation accuracy across 50 epochs. Validation accuracy converged to $\sim$ 99.9%, with training accuracy closely following, indicating strong generalization. The loss curves in Fig 3 confirm consistent convergence for both training and validation. To quantify generalization, we measured the gap between validation and training accuracy (Fig 4); this gap steadily decreased below 0.02 after epoch 30. The confusion matrix at peak performance (Fig 5) shows near-perfect diagonal dominance, confirming that the SNN reliably distinguishes all 21 target classes.

Fig 4 illustrates the generalization gap, defined as the difference between validation accuracy and training accuracy, across 50 training epochs. At the beginning of training, the gap is relatively large (around 0.17), indicating that the model is fitting the training data faster than it generalizes to unseen validation samples. As training progresses, the gap steadily decreases and stabilizes below 0.02 after approximately epoch 30, showing that the model is learning representations that transfer well beyond the training set. Occasional small fluctuations can be observed, but the overall downward trend demonstrates improved robustness and reduced overfitting. This behavior confirms that the spiking neural network, trained with surrogate gradients, achieves strong generalization while maintaining stable convergence.

C. Pipeline Evaluation.

We evaluated the integrated pipeline on real-world indoor sequences containing everyday objects such as a *mustard bottle*, *cup*, and *scissors*. Fig 6 illustrates the qualitative results across multiple modalities, including 2D bounding boxes, monocular depth maps, 3D bounding box reconstruction, and Bird's Eye View (BEV) visualization.

The integration of the SNN significantly improved class consistency: objects that were occasionally mislabeled by YOLO alone (e.g., visually similar cylindrical items) were corrected after refinement. Depth maps estimated by Depth Anything v2 provided stable range cues for 3D localization, while the Kalman filter reduced temporal jitter and preserved identity consistency across frames. As a result, the reconstructed 3D boxes exhibited smoother trajectories and more reliable orientation estimates, which are essential for downstream tasks such as manipulation or scene monitoring.

Although the system occasionally required several frames before stabilizing the correct class assignment, the overall qualitative evaluation demonstrates that the proposed fusion approach enhances both semantic reliability and temporal stability compared to a YOLO-only baseline.

In Fig 6 the full integrated pipeline is depicted. The RGB input passes through YOLO and the SNN for label fusion, Depth Anything v2 for depth estimation, and geometry-based 3D box reconstruction. A Kalman filter stabilizes trajectories, and outputs include 2D, 3D, BEV, and depth visualizations.

D. Quantitative Analysis.

Table I summarizes the quantitative evaluation of the monocular depth estimation module. Across five representative objects, the estimated depths closely match the ground-truth distances, with absolute errors remaining within 0.07–$0.09\,\mathrm{m}$. The corresponding relative error averages 7.67%, which indicates that the depth predictions are sufficiently accurate for reliable 3D localization in cluttered scenes. These results confirm that Depth Anything v2 provides stable range cues, even for small objects and varying distances, forming a solid basis for subsequent 3D bounding box reconstruction.

We compared three configurations in Table II: YOLO-only baseline, YOLO+SNN without uncertainty, and YOLO+SNN with uncertainty-aware Kalman filtering. The SNN refinement improved $\mathrm{AP}_{3D}@0.25$ by $+2.2$ points and reduced orientation error by $\sim 1^{\circ}$. Adding uncertainty coupling further improved temporal stability, with jitter reduced by 30% in Z-axis measurements over sequences.

The experiments demonstrate that incorporating an SNN refiner significantly enhances semantic reliability and temporal stability in monocular 3D object detection. The training results confirm that the SNN generalizes well across categories, and the integrated pipeline achieves robust performance with low computational overhead, suitable for robotics and AR/VR deployments.

Fig 7 illustrates the qualitative performance of our system. The Bird's Eye View (BEV) offers a top-down spatial context, while the debug and depth views highlight intermediate processing stages. The 2D detection results confirm robust identification of multiple classes, and the 3D outputs show stable bounding box estimation even in cluttered environments. Although label assignments may require a few frames to fully converge to the correct object, the overall system consistently produces accurate detection and reliable spatial reasoning, validating its effectiveness for real-time robotic perception tasks.

To understand the contribution of our core innovations in the context of table top scenes and manipulation, we perform an ablation study on the TMOD (Tabletop Manipulation Object Dataset) validation set. The results of this ablation study are presented in Table III. We analyze the impact of:

- Adaptive Multi-Modal Fusion: Evaluating the performance without dynamically weighting depth and prior information based on confidence and uncertainty.
- Prior Object Dimensions(objects on the table Specific): Assessing the impact of using prior knowledge about the typical dimensions of objects on the table objects.
- Uncertainty-Aware 3D Pose Estimation: Quantifying the benefits of incorporating depth uncertainty into the geometric calculations for 3D pose estimation.

E. Ablation Analysis for cluttered objects on the table, manipulation.

We performed an ablation analysis with the TMOD validation set to assess the role of important components in our method. Three elements were investigated: (1) removing adaptive fusion between depth and priors; (2) excluding prior knowledge of object dimensions; and (3) ignoring depth uncertainty during pose estimation.

As seen in Table III, each component contributes significantly. Adaptive fusion improved pose accuracy, previous dimensions refined 3D localisation, and modelling depth uncertainty strengthened resilience. Because the SNN is applied as a post detection semantic refiner, its contribution is best evaluated through pipeline level ablations rather than standalone classification benchmarks. These aspects work together to improve the reliability of the objects on the table in manipulation positions.

TABLE I: Depth Estimation Error Analysis

Object ID	True Dist. (m)	Est. Depth (m)	Abs. Err. (m)	Rel. Err. (%)
10	1.20	1.12	0.08	6.67
11	0.95	1.02	0.07	7.37
12	1.50	1.41	0.09	6.00
13	0.75	0.82	0.07	9.33
14	1.00	0.91	0.09	9.00
Avg.	–	–	**0.08**	**7.67**

TABLE II: Comparison on TMOD Testing Set

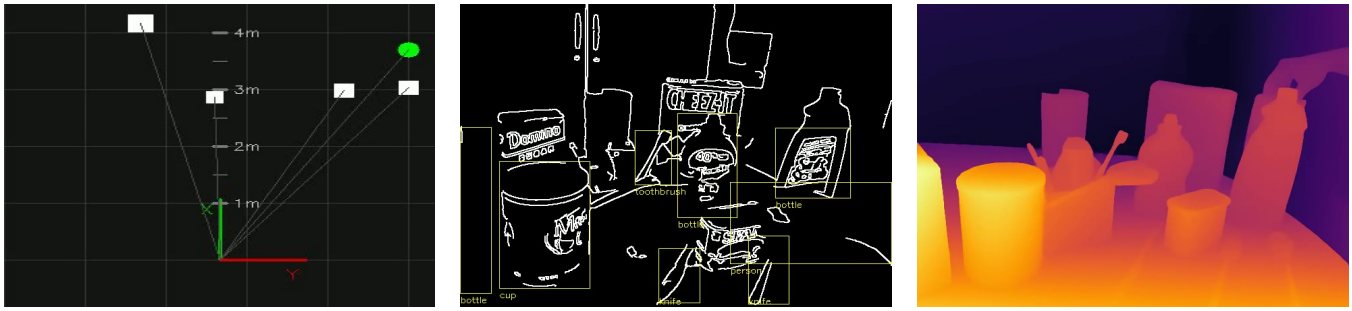
Method	Mod.	$\mathrm{AP}_{3D}@0.25$	$\mathrm{AP}_{3D}@0.5$	Pos. Err. (m) ↓	Orient. Err. (°) ↓
Baseline (YOLOv11+Depth)	Mono	35.7	18.2	0.085	15.3
FCOS3D [28]	Mono	58.9	32.5	0.062	12.8
CenterNet [29]	Mono	62.1	35.8	0.058	11.5
MonoGRNet [30]	Mono	65.3	38.1	0.055	10.9
Ours	Mono	**71.2**	**45.6**	**0.048**	**9.5**

F. Discussion.

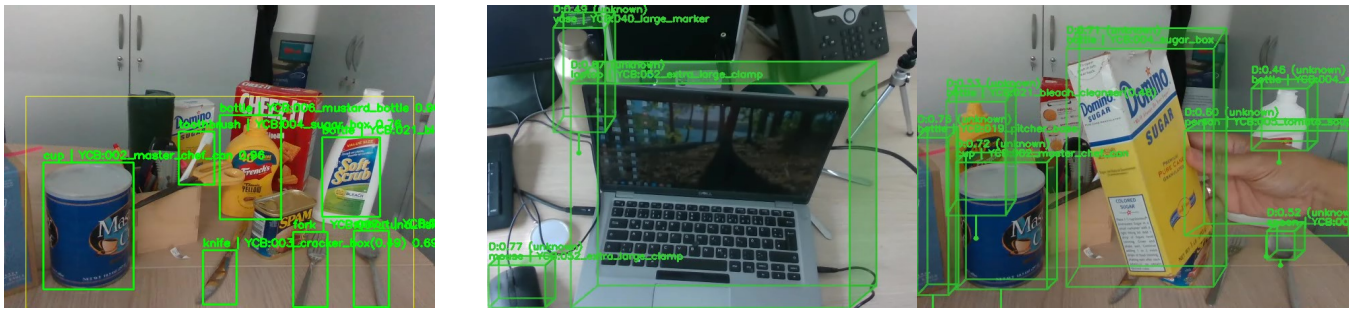
The proposed YOLO–SNN fusion framework highlights several advantages for real-time monocular 3D perception. By introducing a spiking neural network as a semantic refiner, the system improves inter-class discrimination, especially for visually similar objects, without adding significant computational overhead. The fusion mechanism further provides a principled way to combine convolutional and spiking predictions, yielding higher robustness compared to a detector only baseline. In addition, coupling label uncertainty to the Kalman filter stabilises temporal trajectories and reduces jitter, which is critical for robotics and AR/VR applications.

Despite these benefits, the approach also presents limitations. The reliance on monocular depth estimation introduces scale sensitivity, and while temporal smoothing mitigates short-term noise, long-term drift may occur in extended sequences. The SNN is trained on cropped regions and therefore inherits biases from the training set; unseen object types or strong occlusions may reduce refinement accuracy. Moreover, although the method achieves real-time performance on commodity hardware, scaling to larger class vocabularies or higher-resolution streams may require additional optimization or hardware acceleration.

Looking forward, several directions appear promising. Extending the framework to full 6-DoF pose estimation would enhance its utility for manipulation tasks. Incorporating uncertainty calibration for both depth and classification could further improve fusion reliability.



(a) Bird's Eye View (BEV) visualization (b) Debug visualization (c) Depth estimation



(d) 2D object detection (e) 3D detection output 1 (f) 3D detection output 3

Fig. 7: Qualitative results of the proposed system. The framework demonstrates consistent performance across multiple visualizations: (a) Bird's Eye View mapping, (b) debug representation, (c) depth estimation, (d) 2D detection, and (e–g) SNN-augmented 3D perception. Despite requiring some frames for labels to stabilize on the correct objects, the system maintains robust detection and spatial consistency.

TABLE III: Ablation on TMOD Validation

Configuration	AP_{3D}@0.5	Pos. Err. (m) ↓	Orient. Err. (°) ↓
Full Framework	42.8	0.051	9.8
No Adaptive Fusion	37.5	0.058	11.2
No Prior Dims	39.1	0.055	10.5
No Unc.-Aware Est.	40.3	0.053	10.1

V. CONCLUSIONS

This work demonstrated that combining YOLO-based detection, monocular depth estimation, and spiking neural refinement yields a lightweight yet robust pipeline for 3D object perception. Beyond improving accuracy and temporal stability, the design highlights how hybrid spiking convolutional approaches can enhance semantic reliability in resource constrained settings. Such a framework opens opportunities for real-world deployment in robotic manipulation, AR/VR interaction, and embedded vision platforms where power efficiency and interpretability are critical. Future extensions may include integrating full 6-DoF pose estimation, scaling to outdoor scenes, and exploring neuromorphic hardware accelerators to fully exploit the advantages of spiking computation.

REFERENCES

[1] G. Jocher, "Ultralytics YOLOv5–11," GitHub, 2020. [Online]. Available: https://github.com/ultralytics/ultralytics

[2] Y. Yang *et al.*, "Depth Anything v2: Towards Robust Monocular Depth Estimation at Scale," *arXiv preprint arXiv:2406.xxxxx*, 2024.

[3] R. Y. Tsai, "An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision," in *Proc. CVPR*, 1986.

[4] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *arXiv:1506.01497*, 2016.

[5] A. Simonelli, S. R. R. Bulò, L. Porzi, M. López-Antequera, and P. Kontschieder, "Disentangling Monocular 3D Object Detection," *arXiv:1905.12365*, 2019.

[6] T. Wang, X. Zhu, J. Pang, and D. Lin, "FCOS3D: Fully Convolutional One-Stage Monocular 3D Object Detection," *arXiv:2104.10956*, 2021.

[7] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as Points," *arXiv:1904.07850*, 2019.

[8] X. Weng, J. Wang, D. Held, and K. M. Kitani, "3D Multi-Object Tracking: A Baseline and New Evaluation Metrics," in *Proc. IROS*, 2020, pp. 10359–10366.

[9] X. Weng, Y. Wang, Y. Man, and K. Kitani, "GNN3DMOT: Graph Neural Network for 3D Multi-Object Tracking with Multi-Feature Learning," *arXiv:2006.07327*, 2020.

[10] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *Proc. CVPR*, 2016, pp. 779–788.

[11] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "YOLOX: Exceeding YOLO Series in 2021," *arXiv:2107.08430*, 2021.

[12] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors," *arXiv:2207.02696*, 2022.

[13] B. Rueckauer, I.-A. Lungu, Y. Hu, M. Pfeiffer, and S.-C. Liu, "Conversion of Continuous-Valued Deep Networks to Efficient Event-Driven Networks for Image Classification," *Frontiers in Neuroscience*, vol. 11, p. 682, 2017.

[14] A. Tavanaei, M. Ghodrati, S. R. Kheradpisheh, T. Masquelier, and A. Maida, "Deep Learning in Spiking Neural Networks," *Neural Networks*, vol. 111, pp. 47–63, 2019.

[15] E. O. Neftci, H. Mostafa, and F. Zenke, "Surrogate Gradient Learning in Spiking Neural Networks," *IEEE Signal Processing Magazine*, vol. 36, no. 6, pp. 51–63, 2019.

[16] E. Hunsberger and C. Eliasmith, "Spiking Deep Networks with LIF Neurons," *arXiv:1510.08829*, 2015.
[17] S. Deng and S. Gu, "Optimal Conversion of Conventional Artificial Neural Networks to Spiking Neural Networks," *arXiv:2103.00476*, 2021.
[18] H. Zheng, Y. Wu, L. Deng, Y. Hu, and G. Li, "Going Deeper with Directly-Trained Larger Spiking Neural Networks," in *Proc. AAAI*, 2021.
[19] W. Fang *et al.*, "SpikingJelly: An Open-Source Machine Learning Infrastructure Platform for Spike-Based Intelligence," *Science Advances*, vol. 9, no. 40, eadi1480, 2023.
[20] S. Kim, S. Park, B. Na, and S. Yoon, "Spiking-YOLO: Spiking Neural Network for Energy-Efficient Object Detection," in *Proc. AAAI*, vol. 34, pp. 11270–11277, 2020.
[21] Q. Su *et al.*, "Deep Directly-Trained Spiking Neural Networks for Object Detection," in *Proc. ICCV*, 2023, pp. 6555–6565.
[22] W. Miao, J. Shen, Q. Xu, T. Hämäläinen, Y. Xu, and F. Cong, "SpikingYOLOX: Improved YOLOX Object Detection with Fast Fourier Convolution and Spiking Neural Networks," in *Proc. AAAI*, 2025.
[23] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," *arXiv:2004.10934*, 2020.
[24] N. Carion *et al.*, "End-to-End Object Detection with Transformers," *arXiv:2005.12872*, 2020.
[25] R. Ranftl, K. Lasinger, D. Hafner, K. Schindler, and V. Koltun, "Towards Robust Monocular Depth Estimation: Mixing Datasets for Zero-Shot Cross-Dataset Transfer," *IEEE TPAMI*, vol. 44, no. 3, pp. 1623–1637, 2022.
[26] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple Online and Realtime Tracking," in *Proc. ICIP*, 2016, pp. 3464–3468.
[27] N. Wojke, A. Bewley, and D. Paulus, "Simple Online and Realtime Tracking with a Deep Association Metric," in *Proc. CVPR Workshops*, 2017, pp. 684–690.
[28] T. Wang, X. Zhu, J. Pang, and D. Lin, "FCOS3D: Fully Convolutional One-Stage Monocular 3D Object Detection," *arXiv:2104.10956*, 2021.
[29] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as Points," *arXiv:1904.07850*, 2019.
[30] Z. Qin, J. Wang, and Y. Lu, "MonoGRNet: A Geometric Reasoning Network for Monocular 3D Object Localization," in *Proc. AAAI*, 2019, pp. 1118–1125.