HVLO-YOLO : An Ultra-Lightweight Detection Model for High-voltage Line Obstacles

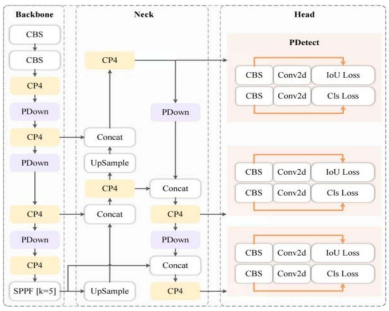
BMVC 2025

Weichao Pan[1], Xu Wang[1], Chengze Lv[1], Zicheng Lin[1], Gongrui Wang[1], Xuening Zhang[2], Yi Sun[3], Xingbo Liu[1, 3, *]
[1] Shandong Jianzhu University, [2] Harbin Institute of Technology, Shenzhen and [3] University of Ulster
*Corresponding author :Xingbo Liu (sclxb@mail.sdu.edu.cn)

Introduction

With the rapid expansion of high-voltage power grids, obstacle detection has become critical to prevent safety hazards from birds, balloons, and debris. Traditional methods based on hand-crafted features struggle in complex environments, while deep learning models demand heavy computation, limiting real-time deployment. The key challenge lies in achieving accurate detection under dynamic scenes and resource constraints. This calls for a solution that balances robustness, efficiency, and lightweight design to ensure reliable obstacle identification in real-world high-voltage scenarios.

Method



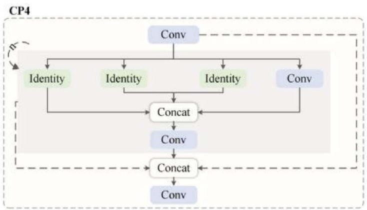
The architecture of HVLO-YOLO follows a classic three-stage object detection pipeline: Backbone, Neck, and Head. To achieve lightweight and efficient detection under real-world constraints, each stage incorporates specialized modules tailored for high-voltage obstacle scenarios.

The Backbone extracts multi-level features using the proposed CP4 and PDown modules, enhancing feature diversity and preserving semantic information while reducing computational cost.

The Neck performs multi-scale feature fusion, where CP4 and PDown further refine and align features across resolutions for robust detection in complex scenes.

The Head uses the lightweight PDetect module to focus computation on informative regions, enabling accurate, real-time predictions with minimal resources.

This modular design strikes an effective balance between detection accuracy and efficiency.



CP4 Module

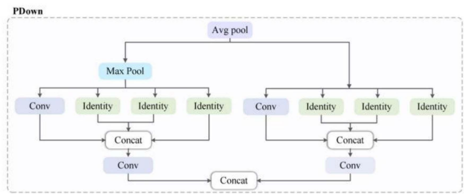
The CP4 module enhances feature extraction by combining partial convolution with group-wise processing in a recursive design. Then, a 1×1 convolution fuses the outputs:

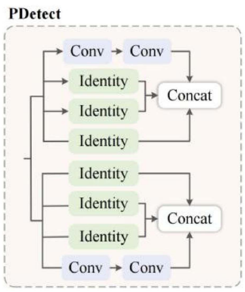
$$Y^{(1)} = W_1 * Concat(W_3 * X^{(1)}, X^{(2)}, X^{(3)}, X^{(4)})$$

To enrich semantics, CP4 recursively applies the same pattern:

$$Y^{(1)} = W_1 * Concat(Y^{(t-1)}, SplitConv(Y^{(t-1)}))$$

This split-selective design ensures feature diversity and efficiency, forming a lightweight yet expressive backbone for HVLO-YOLO.





PDetect module

The PDetect module is a selective and lightweight detection head designed to allocate computation to informative regions only.

Each local aggregation is computed as:

$$Y^{(i)}_{local} = Concat(W_3 * X^{(i)}, X^{(i+1)}, X^{(i+2)}, ...)$$

Final output is fused by a 1×1 convolution:

$$Y = W_1 * Concat(Y^{(1)}_{local}, Y^{(2)}_{local}, ...)$$

For bounding box regression, CIoU loss is used:

$$L_{CIoU} = 1 - (IoU - \frac{\rho^2(b, b^{gt})}{c^2} - \alpha v)$$

This design greatly reduces parameters while maintaining strong localization accuracy.

PDown Module

To achieve efficient downsampling while preserving semantic details, the PDown module employs a dual-path structure:

The input feature map X is first smoothed by average pooling:

$$X_{arg} = \text{Avg Pool}(X)$$

The left path performs max pooling on X_{arg}, followed by partial convolutions across split feature groups:

$$Y_{path1} = W_1 * Concat(W_3 * X^{(1)}_{max}, X^{(2)}_{max}, ...)$$

The right path applies partial convolutions directly to X_{arg}:

$$Y_{path2} = W_1 * Concat(W_3 * X^{(1)}_{arg}, X^{(2)}_{arg}, ...)$$

Final output is the fusion of both paths:

$$Y = Concat(Y_{path1}, Y_{path2})$$

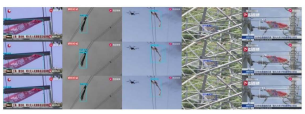
This design ensures both global context and local structure are retained, boosting robustness for small-object detection under complex backgrounds.

Results

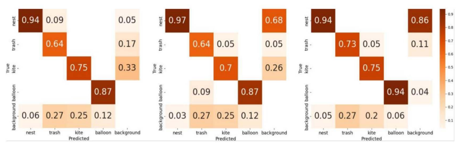
YOLOv9-t

YOLOv10-n

HVLO-YOLO



As shown in the left figure, HVLO-YOLO achieves more precise and complete detections than YOLOv9-t and YOLOv10-n in complex scenes. The improvement comes from CP4, PDown, and PDetect, which enhance representation, retain semantics, and selectively allocate computation to key regions.



As shown in the above figure, HVLO-YOLO significantly outperforms YOLOv9-t and YOLOv10-n, especially in classifying "trash" and "balloon," while reducing background confusion. Its advantage comes from CP4, PDown, and PDetect, which jointly ensure lightweight design and accurate obstacle classification.

Table 1: Detailed comparative results of the contrast experiments. Red font represents baseline models. The rise and fall of the metric is compared to the best baseline models.

Model	P (%)	R (%)	mAP50 (%)	mAP 50:95 (%)	Params (M)	FLOPs (G)	Model Size (MB)
YOLOv8-n	95.7	76.6	84.6	63.1	3.0	8.1	6.3
YOLOv8-s	90.1	81.5	86.8	65.2	11.13	28.4	22.5
YOLOv8-m	92.7	78.0	85.1	65.4	25.84	78.7	52.0
YOLOv9-t	97.0	77.5	88.4	67.1	2.0	7.6	4.7
YOLOv9-s	96.9	80.6	88.4	70.2	6.0	20.8	12.9
YOLOv9-m	96.6	82.0	88.8	70.8	20.0	76.5	40.8
YOLOv10-n	84.7	77.1	82.8	62.8	2.7	8.2	5.8
YOLOv10-s	90.4	80.7	85.3	67.2	7.3	21.2	15.1
YOLOv10-m	89.0	78.7	86.2	67.6	15.7	59.5	31.9
YOLOv10-b	91.2	83.0	88.2	70.9	19.6	93.6	39.8
YOLOv10-l	95.1	77.8	87.4	69.6	24.9	122.0	50.6
YOLOv10-x	**97.1**	80.3	89.8	**71.5**	31.6	169.8	64.1
RT-DETR-l	88.5	**85.9**	87.3	69.7	32.0	103.4	66.2
RT-DETR-x	96.5	77.9	84.5	66.8	65.5	222.5	135.4
RT-DETR-ResNet50	91.9	84.9	87.4	69.6	41.9	125.6	86.0
RT-DETR-ResNet101	92.0	84.1	85.1	69.4	60.9	186.2	124.2
HVLO-YOLO (ours)	91.7 (↓5.3)	82.9 (↑5.4)	**89.9** (↑1.5)	67.6 (↑0.5)	**1.0** (↓1.0)	**2.3** (↓5.3)	**2.2** (↓2.5)

Table 2: Detailed rewe use the number of model param universal object detection.

Model	mAP50 (%)	mAP 50:95 (%)	Params(M)	Model Size(MB)
NanoDet-m	36.5	21.6	0.95	1.8
YOLO-Fastest-1.1-xl	35.4	20.8	0.93	3.7
YOLOv4-t	38.0	23.2	6.1	23.2
MobileDets	42.8	28.1	4.9	13.8
Moblie-ViT-SSDLite	41.1	26.3	2.7	6.3
FastestDet	28.4	14.6	**0.3**	**1.2**
YOLOv5-n	49.1	28.4	1.9	3.9
YOLOv8-n	51.9	37.3	3.2	6.2
YOLOv9-t	52.2	38.3	2.0	4.4
YOLOv10-n	**53.2**	**38.5**	2.3	5.7
HVLO-YOLO (ours)	39.3	25.5	1.0	2.3

Table 3: Detailed results of ablation study.

CP4	PDown	PDetect	P (%)	R (%)	mAP50 (%)	mAP 50-95 (%)	Params(M)	FLOPs(G)	ModelSize(MB)
✓			97.2	82.8	87.0	67.1	2.12	5.9	4.5
✓	✓		89.6	82.3	89.1	65.6	1.04	2.6	2.3
✓		✓	**98.4**	80.2	84.6	64.6	1.54	3.3	3.3
	✓		94.4	**83.1**	88.5	**69.4**	2.47	7.1	5.2
	✓	✓	92.8	80.6	88.1	66.5	1.89	4.4	4.0
		✓	96.7	77.9	86.2	66.1	2.42	5.5	5.1
✓	✓	✓	91.7	82.9	**89.9**	67.6	**1.0**	**2.3**	**2.2**

Conclusion

This paper proposes HVLO-YOLO, an ultra-lightweight object detection framework for high-voltage line obstacles. It incorporates three modules: CP4 for efficient multi-scale feature extraction using partial convolution and FourGroup design; PDown for preserving semantics via dual-branch downsampling; and PDetect for focusing computation on key regions. HVLO-YOLO achieves strong detection accuracy with significantly reduced parameters and FLOPs, outperforming existing models on both high-voltage and COCO datasets. While effective, the model shows limitations in handling extremely small or highly textured objects. Future work will explore adaptive partial convolutions and dcynamic feature selection to further improve robustness and fine-grained detection performance.