Proceedings of the 4th International Conference on Machine Learning and Automation
DOI: 10.54254/2755-2721/2026.39080

DSA-YOLO for Foreign Object and Surface Defect Detection on Mining Conveyor Belts

Maosheng Jia

School of Mechanical Engineering, Anhui University of Science and Technology, Huainan, China
1178698160@qq.com

Abstract. Small foreign objects and elongated defects are difficult to detect in mining conveyor belt images. DSA-YOLO extends YOLOv8n through directional gated residual features, a stride-four branch with prediction towers shared across four scales, and bounded area-balanced regression. Experiments use 2,148 images with 14,700 boxes, acquisition-group isolation, and three random seeds. DSA-YOLO achieves 83.10% mAP50 and 54.63% ± 0.33% mAP50:95, improving on YOLOv8n by 9.47 and 6.36 percentage points. Foreign-object recall reaches 80.06% ± 0.87%. With 3.18 million parameters, mean end-to-end latency is 2.78 ms on an RTX 4090 at FP32 and batch size one, an increase of 0.32 ms. The results support joint detection under the evaluated conditions; transfer to independent sites remains to be assessed.

Keywords: conveyor belt, foreign object detection, surface defect, directional gating, shared head

1. Introduction

Mining conveyor belt inspection must distinguish conveyed material and foreign objects while locating cracks and holes. Existing YOLO-based studies address belt foreign objects [1, 2], but small targets and narrow defects require both spatial detail and shape-sensitive features. Feature pyramids preserve multiscale information [3], while directional encoding provides complementary spatial context [4]. A finer detection branch also increases prediction cost. DSA-YOLO combines directional gated residual modules (DGR), a four-scale shared head (P2-SH), and area-balanced regression (ABR) within YOLOv8n [5].

2. Method

2.1. Directional features and four scale prediction

Figure 1 summarizes the architecture. The YOLOv8n backbone retains C2f and SPPF, with DGR inserted after the stride-8 and stride-16 C2f stages. A bidirectional neck combines backbone levels C2–C5, producing B2–B5 at strides 4, 8, 16, and 32 with 32, 64, 128, and 256 channels. Top-down fusion upsamples C5 through C4, C3, and C2; bottom-up fusion returns from B2 through the coarser levels.

© 2026 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2755-2721/2026.39080

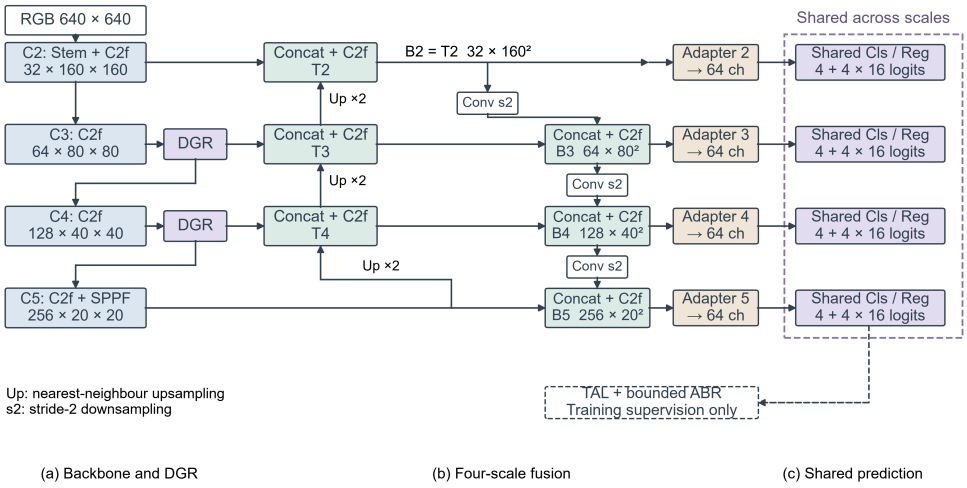

Figure 1. DSA-YOLO architecture. (a) Backbone and DGR. (b) Bidirectional four-scale fusion. (c) Scale-specific adapters and shared prediction. Dashed paths denote training supervision

For an input feature X, DGR computes three bias-free depthwise branches V_j: 3 × 3 local, 1 × 5 horizontal, and 5 × 1 vertical convolutions. A 1 × 1 convolution generates three spatial logits; softmax along the branch dimension produces gates g_j that are broadcast over channels:

$$g = \mathrm{softmax}(\mathrm{Conv}_{1\times1}(X)), Y = X + \beta \mathrm{SiLU}(\mathrm{BN}(\mathrm{PW}(\sum_{j\in\{0,h,v\}} g_j \odot V_j))) \tag{1}$$

PW denotes a pointwise convolution, BN batch normalization, and β a trainable scalar initialized to 0.1. The two modules add 25,096 parameters. Their fixed orthogonal kernels supply directional context without requiring orientation labels or guaranteeing rotation invariance.

P2-SH increases candidate locations from 8,400 to 34,000. Each scale has an independent 1 × 1 adapter followed by eight-group normalization and SiLU, giving 64 channels. Separate classification and regression towers each contain two depthwise-separable blocks (3 × 3 depthwise, 1 × 1 pointwise, group normalization, SiLU). Each tower and its output layer share weights across scales. Outputs contain four class logits and 16 distance bins per box side; expected bin distances are multiplied by the corresponding stride. DFL targets are clipped to [0,14.99]. The head has 54,852 parameters versus 125,712 for equivalent separate towers. Weight sharing reduces parameters; each scale still requires forward computation.

2.2. Bounded area balanced regression

ABR first normalizes assignment quality within each ground-truth object and then weights objects by box area. Let $a_j = w_j h_j/(640 \times 640)$ be the transformed box area. Define

$$r_j = \mathrm{clip}[(\frac{a_0}{\max(a_j,\varepsilon)})^{\gamma_a}, 0.5, 2], a_0 = 0.0025, \gamma_a = 0.5, \varepsilon = 10^{-9} \tag{2}$$

Let q_i be the sum of class target scores assigned to positive candidate i, $J(i)$ its matched object, and P_j the positive set for object j. Across a batch, object indices include the image index. With $M = \{j : m_j > \varepsilon\}$,

DOI: 10.54254/2755-2721/2026.39080

$$m_j = \sum_{i \in P_j} q_i,\ Z = \sum_{j \in M} r_j,\ u_i = \frac{r_{J(i)} q_i}{m_{J(i)} Z} \tag{3}$$

For nonempty M, the candidate weights sum to one, and object j receives total weight r_j/Z, independent of repeated identical candidate sets. Assignment scores and weights are detached from gradients. ABR aggregates the standard complete-IoU (CIoU) and distribution focal losses (DFL) [6, 7]:

$$L_{\text{box}} = \sum_i u_i \ell_{\text{CIoU}}(i),\ L_{\text{dfl}} = \sum_i u_i \ell_{\text{DFL}}(i),\ L = 7.5 L_{\text{box}} + 0.5 L_{\text{cls}} + 1.5 L_{\text{dfl}} \tag{4}$$

Classification retains the baseline soft-target binary cross-entropy and task-aligned assignment [5]. Empty positive batches yield graph-connected zero regression loss. ABR adds no inference operations and does not explicitly rebalance class frequencies.

3. Experiments and discussion

3.1. Data and evaluation protocol

The dataset contains 2,148 RGB frames and 14,700 axis-aligned boxes from four fixed cameras and 36 continuous acquisition groups. Class IDs 0–3 denote block, crack, foreign, and hole. Camera and acquisition period define each group; all associated images remain in one subset to reduce adjacent-frame leakage. Training, validation, and test proportions are approximately 70%, 15%, and 15%, with class coverage considered. Validation selects checkpoints; the test split is fixed.

Images are letterboxed to 640 × 640 and normalized to [0,1]. Training uses horizontal flipping (probability 0.5), scaling ±0.2, translation ±0.1, and rotation ±10°. With probability 0.5, photometric augmentation applies $I' = \text{clip}(gI^{\gamma_p}, 0, 1)$, with $\gamma_p \sim U(0.7, 1.5)$ and $g \sim U(0.8, 1.2)$. Boxes retain at least 20% visible area and two pixels per side. Vertical flipping, Mosaic, and MixUp are disabled; evaluation uses deterministic preprocessing.

All models train for 500 epochs with effective batch size 16 and seeds 0, 1, and 2. SGD uses momentum 0.937, weight decay 0.0005, initial learning rate 0.01, cosine decay to 0.0001, and three warm-up epochs. Matching COCO-pretrained backbone weights initialize each model; the neck and heads are reinitialized. Task-aligned assignment uses top-k 10, $\alpha = 0.5$, and $\beta = 6$. The highest validation mAP50:95 selects the checkpoint, with earlier epochs breaking ties. Software comprises Ubuntu 22.04, PyTorch 2.3.1, CUDA 12.4, and Ultralytics 8.3.0 on an RTX 4090 (24 GB).

AP uses 101-point interpolation with confidence floor 0.001, class-wise NMS IoU 0.7, and at most 300 detections. mAP50 is the equal-weight mean across four classes; mAP50:95 also averages IoU thresholds 0.50–0.95 in 0.05 increments [8]. Recall uses confidence 0.25 and matching IoU 0.5. Small-object AP uses ground-truth area below 32^2 pixels in evaluation coordinates. Tables report three-seed means, with sample standard deviations where available; no significance test is claimed.

3.2. Accuracy and inference cost

Table 1 compares the baseline, a larger YOLOv8s, a P2 extension with independent standard heads, and DSA-YOLO. DSA-YOLO improves mAP50 and mAP50:95 over YOLOv8n by 9.47 and 6.36 percentage points, respectively, and foreign-object recall by 10.22 points. Relative to YOLOv8s and YOLOv8n-P2, mAP50:95 increases by 2.49 and 3.82 points.

DOI: 10.54254/2755-2721/2026.39080

Table 1. Detection accuracy on the fixed test set (%)

Model	mAP50	mAP50:95	Foreign AP50	Foreign recall
YOLOv8n	73.63	48.27 ± 0.41	74.18 ± 0.92	69.84 ± 1.12
YOLOv8s	78.06	52.14 ± 0.38	78.61 ± 0.71	74.29 ± 0.88
YOLOv8n-P2	77.39	50.81 ± 0.46	76.94 ± 0.84	73.51 ± 0.97
DSA-YOLO	83.10	54.63 ± 0.33	82.37 ± 0.61	80.06 ± 0.87

mAP50 is the macro-average of four class AP50 means; ± denotes the sample standard deviation across three seeds.

DSA-YOLO achieves class-wise AP50 values of 89.71%, 78.36%, 82.37%, and 81.94% for block, crack, foreign, and hole. Their mean is 83.10%. Compared with YOLOv8n, gains are 3.29, 16.29, 8.19, and 10.11 points. Improvements therefore extend beyond conveyed material, although ABR itself is based on area rather than category.

Table 2. Model size and synchronized inference cost

Model	Params(M)	GFLOPs	Network(ms)	End to end(ms)	P95(ms)	FPS
YOLOv8n	3.01	8.2	1.84	2.46	2.73	406.5
YOLOv8s	11.14	28.5	2.91	3.61	4.02	277.0
YOLOv8n-P2	3.48	13.6	2.06	2.70	2.98	370.4
DSA-YOLO	3.18	14.1	2.14	2.78	3.08	359.7

RTX 4090, FP32, batch = 1; 100 warm-up and 1,000 synchronized runs. Latencies are means except P95. FLOPs = 2 MACs.

End-to-end timing includes preprocessing, transfer, forward inference, and NMS; network timing covers only forward inference. FPS equals 1,000 divided by mean end-to-end milliseconds. DSA-YOLO has median latency 2.73 ms. Its mean increases by 0.32 ms (13.0%) over YOLOv8n and 0.08 ms over YOLOv8n-P2, while remaining 0.83 ms faster than YOLOv8s (Table 2). The P2-related FLOP increase does not translate proportionally into wall-clock latency on this GPU; other devices require measurement.

3.3. Module ablations and sensitivity

Table 3. Module ablations under the same training protocol

Configuration	mAP50:95(%)	Small AP(%)	Foreign recall(%)	Params(M)
Baseline	48.27	31.42	69.84	3.01
A	49.16	32.08	70.62	3.04
B	52.18	38.74	75.41	3.16
C	49.85	34.81	71.19	3.01
AB	53.41	40.17	77.03	3.18
AC	50.73	35.62	72.38	3.04
BC	53.28	41.09	77.84	3.16
ABC	54.63	42.58	80.06	3.18
P2 only	50.81	38.09	73.51	3.48

DOI: 10.54254/2755-2721/2026.39080

Table 3. (continued)

Shared head only	48.51	31.77	70.08	2.94

A: DGR; B: P2-SH; C: ABR. P2 only is the same configuration and experiment as YOLOv8n-P2. Shared head only retains three scales.

In Table 3, A, B, and C individually raise mAP50:95 by 0.89, 3.91, and 1.58 points. Removing each from ABC lowers accuracy by 1.35, 3.90, and 1.22 points, respectively. P2-SH outperforms P2 only by 1.37 points with 0.32 million fewer parameters. This comparison changes tower composition and normalization alongside sharing, so it does not isolate weight sharing alone. Full-model small-object AP reaches 42.58% versus 31.42% for the baseline.

One-factor sensitivity experiments favor $\gamma_a = 0.5$ and kernel length 5 in validation. Setting $\gamma_a = 0$ retains within-object normalization and yields 53.21% test mAP50:95, below AB without ABR (53.41%); normalization alone is thus not independently supported as beneficial. Width 96 achieves higher validation/test mAP50:95 (55.67%/54.88%) than width 64 (55.41%/54.63%), but increases parameters from 3.18 to 3.31 million and latency from 2.78 to 2.94 ms. Width 64 is a cost–accuracy compromise.

4. Conclusion

DSA-YOLO combines directional features, four-scale shared prediction, and bounded area-balanced localization supervision. On the evaluated dataset, it achieves 54.63% mAP50:95 and 80.06% foreign-object recall with 3.18 million parameters and 2.78 ms mean end-to-end latency. Acquisition-group isolation reduces temporal leakage, but four cameras do not establish transfer to unseen mines or lighting conditions. Axis-aligned boxes also do not measure crack connectivity or physical defect size. Independent-site and field-response evaluation remain necessary.

References

[1] Zhang, Z., Tu, F., Chen, X., et al. (2025). Research on foreign object detection technology for high-magnetite conveyor belts based on YOLOv8 [in Chinese]. *Gold Science and Technology, 33*(1), 193–201. doi:10.11872/j.issn.1005-2518.2025.01.214.

[2] Xiao, D., Liu, P., Wang, J., Gu, Z., & Yu, H. (2023). Mining belt foreign body detection method based on YOLOv4_GECA model. *Scientific Reports, 13*, 8881. doi:10.1038/s41598-023-35962-3.

[3] Lin, T.-Y., Dollár, P., Girshick, R., et al. (2017). Feature pyramid networks for object detection. In *CVPR* (pp. 2117–2125).

[4] Hou, Q., Zhou, D., & Feng, J. (2021). Coordinate attention for efficient mobile network design. In *CVPR* (pp. 13713–13722).

[5] Ultralytics. (2024). *Ultralytics YOLO*, version 8.3.0 [software]. Model configuration, loss functions and task-aligned assignment. https://github.com/ultralytics/ultralytics/tree/v8.3.0 (accessed October 16, 2026).

[6] Zheng, Z., Wang, P., Liu, W., et al. (2020). Distance-IoU loss: Faster and better learning for bounding box regression. In *AAAI, 34*(7), 12993–13000. doi:10.1609/aaai.v34i07.6999.

[7] Li, X., Wang, W., Wu, L., et al. (2020). Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. In *Advances in Neural Information Processing Systems, 33*, 21002–21012.

[8] Lin, T.-Y., Maire, M., Belongie, S., et al. (2014). Microsoft COCO: Common objects in context. In *ECCV* (pp. 740–755).