A Controlled Benchmark of Active Learning Strategies for Class-Imbalanced Waste Detection on SortWaste with YOLO

Thanh Thong Le[1], Tuyet Nhi Chu[2], Manh Hung Nguyen[3]*
Faculty of Information Technology,
Posts and Telecommunications Institute of Technology (PTIT), Hanoi, Vietnam[1,2,3]
Intelligent Computing for Sustainable Development Laboratory (IC4SD),
Posts and Telecommunications Institute of Technology (PTIT), Hanoi, Vietnam[3]

***Abstract*—Bounding-box annotation constrains waste detection under class imbalance and video redundancy. This benchmark compares Random, Plug and Play Active Learning (PPAL), Rarity-Aware, and an entropy/diversity strategy (Entropy) on SortWaste with YOLOv8n over six paired seeds and 5%–20% labeled-image budgets. The primary metric is normalized area under the budget curve (nAUBC) of mean average precision (mAP@50:95) across intersection-over-union (IoU) thresholds of 0.50–0.95. Mean nAUBC $\pm$ sample standard deviation was 0.2831 ± 0.0079 for PPAL, 0.2780 ± 0.0063 for Entropy, 0.2705 ± 0.0057 for Random, and 0.2661 ± 0.0057 for Rarity-Aware. At 20%, mean test mAP@50:95 was 0.3000, 0.2967, 0.2917, and 0.2817, respectively. The paired 20% differences from Random had intervals that included zero for PPAL, Entropy, and Rarity-Aware. Mean paired nAUBC differences from Random were $+0.0126$ for PPAL (unadjusted 95% paired t-interval [0.0032, 0.0220]) and $+0.0075$ for Entropy ([0.0020, 0.0129]). Each seed–budget point was trained once, and the intervals are not adjusted for multiple comparisons. PETOil contributed 46% and 53% of the PPAL and Entropy nAUBC gains. Active strategies selected images with more boxes and greater temporal concentration; their mean gains were smaller under a box-count budget.**

***Keywords*—Active learning; object detection; YOLO; class imbalance; waste detection; image-budget efficiency**

I. Introduction

Waste streams contain heterogeneous materials, including plastics, cardboard, and metals. Object detection can support sorting and monitoring by identifying each object's category and location [1].

Detectors in the You Only Look Once (YOLO) family predict an object's category and location through a bounding box [2]. Waste detection remains data-intensive because bounding-box annotation is costly and material classes are unevenly represented.

Object detection requires annotators to label both the category and spatial extent of each object. Annotation effort therefore depends on image count and objects per image. In a densely annotated dataset such as SortWaste [1], equal image budgets can correspond to different annotation volumes. Cumulative ground-truth box counts are therefore reported alongside labeled-image counts.

*Corresponding author.

Waste datasets may exhibit long-tailed class distributions [3]. In this training split, the most frequent class has approximately 21 times as many instances as the least frequent class. Detectors therefore receive less supervision for lower-count classes, whose remaining errors may depend on relatively few or ambiguous examples. Uniform sampling preserves the pool distribution in expectation but does not target model uncertainty or class scarcity.

Active learning addresses these constraints through selective annotation [4]. A detector trained on a small labeled set repeatedly selects batches from the unlabeled pool. The acquisition function shapes the growing set's class composition, annotation density, and redundancy.

Published object-detection acquisition results use different datasets, detectors, training histories, and annotation protocols, which limits direct comparison [5], [6]. Controlled evidence is especially relevant to industrial waste data because scenes are dense, class distributions are imbalanced, and adjacent video frames can be similar.

This benchmark evaluates four acquisition strategies on SortWaste [1] using the nano variant of YOLOv8 (YOLOv8n) [2], initialized from Common Objects in Context (COCO) weights [7]. The strategies are Random Sampling, Plug and Play Active Learning (PPAL) [8], Rarity-Aware Stratified Sampling [9], and an entropy/diversity pipeline combining Entropy-based Non-Maximum Suppression (ENMS) with Diversity Prototypes (DivProto), referred to as Entropy [10]. The dataset split, initial checkpoint, training configuration, six paired seeds, and cumulative labeled-image budgets are shared; the acquisition strategy and detector adapter vary. The evaluation covers 5%–20% of the 3,705-image training split. Analyses compare trajectory and endpoint performance, lower-count-class average precision (AP) and exposure, annotation volume, and temporal concentration in the acquired images.

II. Background and Related Work

A. Active Learning for Object Detection

Pool-based active learning assumes an initial labeled set and a larger unlabeled pool [4]. At acquisition round t, a detector trained on the current labeled set produces predictions and representations for the unlabeled images. An acquisition

function then selects a query batch for annotation, after which the labeled and unlabeled sets are updated.

For object detection, the value of an image cannot be described by a single class probability. An image may contain multiple objects, predictions may differ in both classification and localization quality, and several detections or several frames may convey nearly identical information. Acquisition strategies therefore combine uncertainty, diversity, coverage, or class allocation in different ways.

Loss prediction estimates the training loss of unlabeled samples [11], while localization-aware acquisition uses box tightness and stability [12]. Probabilistic methods model classification and localization uncertainty [13]; consistency-based methods compare predictions from original and augmented images [14]. The benchmark evaluates four acquisition strategies under a shared detector and training protocol, with analyses of class exposure, annotation volume, and temporal concentration.

B. Class Imbalance in Object Detection

Class imbalance occurs when object instances are distributed unevenly across categories [3]. In detection, imbalance has several dimensions: boxes per class, images containing each class, objects per image, and within-class visual diversity. A low instance count does not necessarily mean that a class is absent from most images. Likewise, repeatedly selecting images containing a lower-count class does not guarantee that the selected examples resolve its remaining detection errors.

The distinction is especially important under macro-averaged mAP: every class contributes equally to the final score, regardless of evaluation support. A large improvement in one lower-count class can therefore dominate the aggregate difference between strategies. Additional labeled instances may still fail to improve AP when examples are redundant, ambiguous, or incorrectly routed by the current detector. A useful benchmark therefore reports both class exposure and class-level performance.

C. Acquisition Strategies

1) Random sampling: Random Sampling selects images uniformly without replacement. It does not use detector uncertainty, image features, or class statistics. Because it introduces no model-dependent preference, it provides the essential baseline for determining whether a more complex acquisition function adds value beyond cumulative coverage of the pool.

2) Plug and play active learning: PPAL separates acquisition into difficulty estimation and coverage [8]. Difficulty-Calibrated Uncertainty Sampling (DCUS) uses training-derived class quality to weight uncertain detections. Category-aware Coverage Maximization Sampling (CCMS) selects a representative subset from the difficult candidate pool using relationships between detection-level class predictions. Together, these stages target images that expose unresolved model error and select a subset that covers this difficult region without excessive repetition.

3) Rarity-aware stratified sampling: Rarity-Aware sampling estimates which predicted classes are underrepresented, increases their acquisition weight, and reserves part of the query for class-conditioned quotas [9]. Its composite image score combines uncertainty, information, confidence, geometric variation, and embedding coverage. The acquisition strategy directly addresses long-tailed data, but its effectiveness depends on whether predicted class presence is reliable and whether the fixed ranking within a query batch preserves useful variation.

4) Entropy, ENMS, and DivProto: The Entropy pipeline begins with predictive entropy [10]. ENMS suppresses redundant detections within each image, and DivProto is applied during batch construction. Class prototypes are updated after each accepted image, so the value of remaining candidates changes during acquisition. A quota mechanism directs most of the theoretical budget toward the least represented half of the classes.

D. Summary

The strategies use four types of evidence: classification uncertainty, localization difficulty, class rarity, and feature-space diversity. Predictive or binary class entropy measures uncertainty. PPAL estimates localization difficulty from training-derived confidence–IoU quality. Rarity-Aware weights predicted classes using pool frequency and labeled-set counts. CCMS, embedding coverage, and prototype selection provide diversity information. Each strategy applies these signals through its own acquisition rule.

Binary class entropy is highest at a probability of 0.5 and approaches zero near 0 or 1. Low confidence–IoU quality indicates localization difficulty; lower predicted class frequency increases rarity weighting; and greater feature-space distance from selected images indicates diversity.

III. Benchmark Methodology

A. Research Questions

The benchmark addresses three research questions.

RQ1. Which acquisition strategy achieves the strongest detection performance under a limited labeled-image budget?

Image-budget efficiency is assessed across the full 5%–20% trajectory using nAUBC of test mAP@50:95. Test mAP@50:95 at 20% is reported as a complementary endpoint measure.

RQ2. Do the acquisition strategies improve the AP of lower-count classes?

This question examines whether active learning improves lower-count classes or whether frequent classes account for most of the aggregate mAP gain. AP is analyzed separately for every class, with particular attention to the three classes containing the fewest training instances: PETOil, Metal, and Cardboard. The number of bounding boxes added to the labeled set for each class is also tracked after every acquisition round.

RQ3. How does sample-selection behavior relate to the efficiency observed in the benchmark?

Beyond detector scores, the benchmark analyzes the properties of the selected images. These properties include the number of objects per image, class distribution, number of bounding boxes requiring annotation, and redundancy among neighboring video frames. The analysis examines whether performance differences coincide with object density, class composition, or repeated visual content in the selected images.

B. Active Learning Protocol

The six paired seeds were 13, 42, 1337, 2718, 31415, and 65536. All six were listed in the project configuration before any run, hence before the composition of any initial set could be inspected. Seeds 13, 42, and 1337 were used in the initial submission. Seeds 2718, 31415, and 65536 were added during revision, and the runs for the first three seeds were retained. For each seed, 185 images (5% of the training split) were sampled uniformly without replacement; class annotations were not used during sampling. Every strategy received the same seed-specific initial labeled set. Cumulative budgets were 5%, 7.5%, 10%, 12.5%, 15%, 17.5%, and 20%, corresponding to 185, 278, 370, 463, 556, 648, and 741 labeled images. At each round, the current detector scores only the unlabeled pool. Selected images are fully annotated and removed from the pool before the next round. Acquisition functions never access ground-truth labels in the unlabeled pool.

After sampling, ground-truth labels were used only for a prespecified validity gate. Each initial set had to contain at least 20 images and 30 boxes for every class, at least 2,500 boxes in total, and no more than 5% background-only images. A failed seed would terminate the run; neither its image set nor the seed would be replaced. All six seeds passed. This gate prevents unusable low-budget training but does not create a class-balanced or oracle-stratified initialization. At round t, L_t and U_t denote the labeled set and unlabeled pool. The detector is f_{θ_t}, with parameters θ_t; K_t is the acquisition batch size, $\mathcal{A}$ the acquisition function, and Q_t the selected query batch. Following the pool-based active learning formulation of Settles [4], the query and set updates are given by Eq. (1) and (2):

$$Q_t = \mathcal{A}(U_t, f_{\theta_t}, L_t; K_t), \tag{1}$$

$$L_{t+1} = L_t \cup Q_t, \qquad U_{t+1} = U_t \setminus Q_t. \tag{2}$$

Thus, Q_t contains exactly K_t distinct images from the current unlabeled pool. Validation and test results are recorded for evaluation only and are never used by an acquisition function. Acquisition stops after the final budget has been evaluated. Full-image annotation adds all ground-truth boxes of each selected image to the labeled set.

C. Implementation of Acquisition Strategies

The equations define the strategy-specific scores, whereas Algorithm 1 specifies the executable acquisition order. Candidate pools contain distinct unlabeled images. Unless stated otherwise, score rankings use decreasing score followed by ascending image path as the deterministic tie-breaker. Seeded randomness is used only where explicitly indicated, and all draws share one persistent random number generator (RNG) per seed. Greek parameters are defined separately within each strategy; the symbols α and β denote unrelated quantities in the PPAL, Rarity-Aware, and Entropy descriptions. Unlabeled ground-truth annotations are never accessed during acquisition. Suppression settings below refer to non-maximum suppression (NMS).

Algorithm 1 Unified active-learning loop and acquisition procedure

Require: Training pool D_{tr}, budget fractions $(b_0, \ldots, b_R)$, strategy a, seed s
Ensure: Detector results and labeled sets $L_0, \ldots, L_R$
Initialize persistent RNG state from s; sample L_0 uniformly without replacement, with $|L_0| = \mathrm{round}(b_0|D_{\mathrm{tr}}|)$
for $t = 0, \ldots, R$ **do**
  Train f_{θ_t} on L_t; evaluate fixed validation/test splits
  if $t = R$ **then**
    break ▷ final budget
  end if
  $C_t \leftarrow \mathrm{sort}(\mathrm{unique}(D_{\mathrm{tr}} \setminus L_t))$
  $K_t \leftarrow \mathrm{round}(b_{t+1}|D_{\mathrm{tr}}|) - |L_t|$
  $Q_t \leftarrow \textsc{Acquire}(a, C_t, K_t, f_{\theta_t}, L_t, s)$
  Assert $|Q_t| = K_t$, $|\mathrm{unique}(Q_t)| = K_t$, $Q_t \cap L_t = \varnothing$
  $L_{t+1} \leftarrow \mathrm{sort}(L_t \cup Q_t)$
end for
procedure $\textsc{Acquire}(a, C, K, f_\theta, L, s)$
  if $K = 0$ **then**
    return $\varnothing$
  end if
  if $a = \textsc{Random}$ **then**
    $Q \leftarrow \varnothing$; $Q \leftarrow \textsc{SampleWithoutReplacement}(C, K, \mathrm{RNG})$
  else if $a = \mathrm{PPAL}$ **then**
    DCUS: conf. 0.05, NMS IoU 0.7, max. 200 detections/image
    $C_4 \leftarrow$ top $\min(4K, |C|)$ by $(-\mathrm{DCUS}, \mathrm{path})$; $V \leftarrow$ retained-detection images
    $Q \leftarrow \varnothing$; $Q \leftarrow \textsc{KMedoidsCCMS}(V, \min(K, |V|), \mathrm{RNG})$; zero-detection images are excluded from CCMS
    if $|Q| < K$ **then**
      $Q \leftarrow Q \cup \textsc{SampleWithoutReplacement}(\mathrm{sort}(C_4 \setminus V), K - |Q|, \mathrm{RNG})$
    end if
  else if $a = \textsc{Rarity-Aware}$ **then**
    $Q \leftarrow \varnothing$
    Rank composite scores (embedding k-means seed drawn from RNG); compute predicted frequencies and effective class weights
    $K_{\mathrm{strat}} \leftarrow \lfloor \rho K \rfloor$; round quotas to nearest integer, adjust by residual weight/class-ID ties; if no classes are predicted, use global fill
    Process classes by $(-\mathrm{weight}, \mathrm{classID})$; dominant-class then presence fallback
    Redistribute unavailable positions by weight; fill remaining positions by global score
  else if $a = \textsc{Entropy}$ **then**
    $Q \leftarrow \varnothing$
    Rank ENMS entropy by $(-\mathrm{entropy}, \mathrm{path})$
    Target lowest-count αn_c classes; count ties use class index; set $q_c = \frac{\beta}{\alpha n_c} K$
    Scan ranked candidates while $|Q| < K$ and active quotas remain
    Reject if $\mathrm{intraSim}(x, Q) \geq 0.7$; accept only if some active c has $\mathrm{presence}_c(x) > 0.3$
    Decrease every corresponding q_c; deactivate $q_c \leq 0$; after quota/list exhaustion, fill remaining positions by global entropy rank
  end if
  Stop when $|Q| = K$; assert $|Q| = K$ and $Q \cap L = \varnothing$
  return $\mathrm{sort}(Q)$
end procedure

1) Random sampling: Random Sampling draws K images uniformly without replacement from the sorted unlabeled pool using the experiment seed. It has no confidence threshold, feature extractor, or class allocation rule.

2) PPAL: The PPAL adapter [8] first estimates class difficulty from positive training assignments indexed by j. Assignment quality q_j combines confidence $p_{j,\mathrm{gt}}$ in the assigned ground-truth class with the localization overlap between boxes b_j and $\hat{b}_j$, as defined in Eq. (3):

$$q_j = p_{j,\mathrm{gt}}^{\xi}\,\mathrm{IoU}(b_j,\hat{b}_j)^{1-\xi}, \qquad \xi = 0.6. \tag{3}$$

The fixed exponent $\xi = 0.6$ slightly favors classification confidence while retaining localization quality. For batch k and class c, let $\mathcal{P}_c^{(k)}$ denote the set of positive assignments, $\bar{q}_c^{(k)}$ their mean assignment quality, $Q_c^{(k)}$ the class-quality exponential moving average (EMA), and $m_c^{(k)}$ its effective momentum. The implementation uses a base momentum of 0.99 and progressively decreases the effective momentum when a class has no positive-quality observation. Following the PPAL update rule [8], the implemented recurrence is given by Eq. (4), (5), and (6):

$$\bar{q}_c^{(k)} = \frac{\sum_{j\in\mathcal{P}_c^{(k)}} q_j}{|\mathcal{P}_c^{(k)}| + 10^{-5}}, \tag{4}$$

$$Q_c^{(k)} = m_c^{(k)} Q_c^{(k-1)} + \left(1 - m_c^{(k)}\right)\bar{q}_c^{(k)}, \tag{5}$$

$$m_c^{(k+1)} = \begin{cases} 0.99, & \bar{q}_c^{(k)} > 0, \\ 0.99 m_c^{(k)}, & \bar{q}_c^{(k)} = 0 \end{cases}. \tag{6}$$

The resulting class quality then scales detection uncertainty as specified in Eq. (7) and (8) [8]:

$$w_c = 1 + \alpha \ln\left[\left(e^{1/\alpha} - 1\right)(1 - Q_c) + 1\right] u_b, \tag{7}$$

$$\alpha = 0.3, \qquad u_b = 0.2, \qquad 1 \le w_c \le 1.2. \tag{8}$$

The curvature $\alpha = 0.3$ controls how rapidly low class quality increases the weight, while the upper-boost parameter $u_b = 0.2$ limits the class weight w_c to the interval $[1, 1.2]$. Image difficulty is the sum of class-weighted binary entropy over retained detections. The confidence threshold is 0.05, the NMS IoU threshold is 0.7, and the DCUS limit is 200 detections per image. The top δK images form the difficult candidate pool, with $\delta = 4$.

CCMS retains at most 100 detections per candidate. The YOLOv8 adapter normalizes independent sigmoid class scores, matches detections without forcing identical predicted labels, and uses symmetrized directional Kullback–Leibler (KL) distance between matched class distributions. K-medoids uses greedy k-center initialization and at most 100 iterations. Images with no retained detections receive zero DCUS score and are excluded from CCMS; seeded random fill is used only when the valid candidate set is insufficient, as specified in Algorithm 1.

3) Rarity-aware stratified sampling: Rarity-Aware sampling [9] estimates class rarity from predicted detection frequencies in the unlabeled pool. Its amplification factor α_c for class c is defined in Eq. (9):

$$\alpha_c = \mathrm{clip}\left[\left(\frac{\tilde{f}}{f_c + \varepsilon}\right)^{\beta}, \alpha_{\min}, \alpha_{\max}\right]. \tag{9}$$

Here, $\tilde{f}$ is the median predicted frequency, f_c is the predicted frequency of class c, and $\beta = 0.6$ controls the strength of rarity amplification. The bounds $\alpha_{\min} = 0.8$ and $\alpha_{\max} = 2.5$ prevent vanishing or extreme class weights, while $\varepsilon = 10^{-12}$ provides numerical stability. A focal exponent $\gamma = 2$ suppresses confident predictions so that uncertainty is concentrated on unresolved detections. The five image components—diversity, information, confidence, geometry, and embedding coverage—are independently max-normalized and combined with equal weight, avoiding an additional tuned coefficient.

Half of each query is routed through class-conditioned allocation ($\rho = 0.5$), with remaining positions filled by global score. Integer quotas account for predicted rarity and labeled-class counts; unavailable positions are redistributed by class weight before global filling. The amplification and quota parameters follow the source severe-imbalance setting [9]. The YOLOv8 adapter uses confidence 10^{-4}, NMS IoU 0.7, and a 300-detection cap; the cap follows the source retention setting without assuming equivalent confidence calibration across detectors. Image embeddings are clustered with k-means++ into K clusters using at most 100 iterations. Source-based method parameters and fixed detector-adapter settings were specified before evaluation; strategy-specific tuning was not performed for any acquisition strategy. All strategies restart from pretrained weights at every budget, whereas the native Rarity-Aware study uses warm continuation.

4) Entropy-ENMS-DivProto: For a retained class probability p, binary predictive entropy $h(p)$ is defined in Eq. (10) [10]:

$$h(p) = -p\ln p - (1-p)\ln(1-p). \tag{10}$$

Predictions use confidence 0.05, NMS IoU 0.7, and at most 300 detections. Each retained detection is associated with a 64-dimensional feature vector taken from the input to the YOLO Detect head and aligned with the post-NMS box. ENMS suppresses redundant same-class detections within an image at similarity threshold 0.5. Lower values would suppress more aggressively; higher values would retain more repeated evidence. ENMS changes the image uncertainty score, whereas class prototypes are calculated as entropy-weighted means over all post-NMS detections of the class.

DivProto constructs the query progressively. The intra-class threshold is 0.7 and the inter-class threshold is 0.3. The least represented α = 0.5 fraction of classes—four of the eight classes—forms the minority set. The theoretical minority allocation is β = 0.75 of the batch, implemented as a per-class quota of $0.1875K$. If an accepted image satisfies multiple active minority classes, each corresponding quota is

decreased; exhausted quotas are removed from the active set. If quota positions remain unavailable, the global entropy ranking supplies the final fill. Images with no retained detections receive zero entropy and can enter only through the final fill.

IV. Evaluation

A single controlled benchmark setup supports the three complementary research questions. Section IV-A describes the shared dataset, protocol, and metrics. Sections IV-B–IV-D address overall image-budget performance (RQ1), lower-count-class performance (RQ2), and selection behavior and annotation efficiency (RQ3), respectively. Section IV-E combines the evidence needed to explain changes in the rankings across budgets, classes, and annotation-workload views.

A. Common Experimental Setup

1) Dataset: Experiments use SortWaste [1], a densely annotated industrial waste-detection dataset containing 5,261 images. The official split provides 3,705 training images, 780 validation images, and 776 test images. SortWaste divides the recorded videos into scenes of 200 consecutive frames, discards the first five frames of each scene, and assigns whole scenes to the three subsets [1]. Both source videos contribute to the training, validation, and test sets. The training set contains 61,842 annotated objects across eight material categories: HDPE (16,803), ECAL (13,649), PET (11,976), Mixed Soft Plastic (9,077), Mixed Rigid Plastic (7,066), Cardboard (1,524), Metal (945), and PETOil (802). The resulting maximum-to-minimum instance ratio is approximately 20.95. Four background-only training images are retained.

SortWaste directly stresses three assumptions of the acquisition functions. Its long-tailed instance distribution permits evaluation of whether explicit class-aware selection improves lower-count classes. Dense multi-object images make image count an imperfect proxy for annotation volume, while consecutive frames from the two video sequences can introduce substantial visual and temporal redundancy. These properties allow analysis of strategy rankings, the factors behind them, and the resulting annotation volume.

These properties also delimit interpretation. Because both videos appear in every subset, the split does not evaluate transfer to an unseen conveyor or video source. Box-level class frequency differs from image-level class presence, and dense or temporally concentrated selections can receive high scores. Class composition, cumulative box count, and short-range frame proximity are analyzed as properties of selected data, not as acquisition inputs.

2) Experimental protocol: Table I consolidates the shared configuration and strategy-specific acquisition settings.

Each strategy used the same COCO-pretrained YOLOv8n checkpoint [2], [7] and shared training and evaluation settings (Table I). Every budget was trained independently from the pretrained weights; the final checkpoint supplied evaluation predictions and the next acquisition. Detector, initialization, optimization, evaluation, and test-prediction protocol were held constant, while the acquisition strategy and its adapter varied. The six paired seeds and shared initial labeled sets are specified in Section III-B. Each saved round checkpoint was evaluated on the test split in one pass. The reported test metrics come from this pass and differ from the values logged during training by less than 0.0003 mAP@50:95.

TABLE I. Benchmark Configuration. Shared Settings Apply to All Six Seeds and Budgets; Strategy-Specific Settings are Listed Separately

Configuration group	Setting
Detector / initialization	YOLOv8n; same COCO-pretrained checkpoint at every budget; no warm start.
Training	640×640; batch 64; 100 epochs/budget; automatic mixed precision (AMP) training; 32-bit floating-point (FP32) evaluation.
Optimizer / schedule	AdamW; learning rate 8.33×10^{-4}, final multiplier 0.01 (linear decay); weight decay 5×10^{-4}; no warmup or early stopping; `close_mosaic` $= 0$.
Augmentation	Mosaic 1.0; MixUp/CutMix 0; horizontal flip 0.5; hue, saturation, and value (HSV) $(0.015, 0.7, 0.4)$; translation 0.1; scale 0.5; rotation/shear/perspective 0.
Evaluation / checkpoint	Confidence 0.001; NMS IoU 0.7; max. 300 detections/image; no test-time augmentation; `last.pt` for evaluation and acquisition.
Protocol / budget	Official SortWaste split; budgets 5%–20%, step 2.5%; initial uniform 5% (185 images), checked by the validity gate in Section III-B.
Seeds	13, 42, 1337, 2718, 31415, 65536.
Acquisition inference	Non-random strategies: input 640; batch 32; FP32; NMS IoU 0.7; no test-time augmentation.
Random	Uniform sampling without replacement; no model inference.
PPAL	Confidence 0.05; detection caps: DCUS 200, CCMS 100; $\xi = 0.6$, $\alpha = 0.3$, $\delta = 4$, $u_b = 0.2$; EMA momentum 0.99; CCMS max. 100 iterations.
Entropy	Confidence 0.05; max. 300 detections/image; ENMS similarity 0.5; intra-/inter-class thresholds 0.7/0.3; minority fraction $\alpha = 0.5$, allocation $\beta = 0.75$.
Rarity-Aware	Confidence 10^{-4}; max. 300 detections/image; rarity bounds $[0.8, 2.5]$; $\beta = 0.6$, focal exponent $\gamma = 2$, stratified fraction $\rho = 0.5$, $\varepsilon = 10^{-12}$.
Software	Python 3.12.13; Ultralytics 8.4.51; PyTorch 2.10.0+cu128; CUDA 12.8.
Hardware	NVIDIA Tesla T4; one graphics processing unit.

FP32 evaluation includes validation and testing.

3) Evaluation metrics: The primary curve-level metric is nAUBC over the realized labeled-image interval from 5% to 20%, following the area-under-budget construction in [15] and [16]. Let b_r and m_r denote the realized labeled fraction and corresponding test mAP@50:95 at budget point r, where, $r = 0, \ldots, R$ and R is the final budget-point index. Trapezoidal integration is computed with Eq. (11) and (12):

$$A = \sum_{r=0}^{R-1} \bar{m}_r \Delta b_r, \qquad \bar{m}_r = \frac{m_r + m_{r+1}}{2}, \tag{11}$$

$$\Delta b_r = b_{r+1} - b_r, \qquad \mathrm{nAUBC} = \frac{A}{b_R - b_0}. \tag{12}$$

Normalization leaves nAUBC on the same scale as mAP. A strategy receives a high value when it improves the detector early and maintains that advantage across subsequent budgets. The final mAP@50:95 at 20% is reported separately because it measures the detector at one operating point. A high nAUBC

but similar endpoint indicates an early advantage that later narrows; a strong endpoint with lower nAUBC indicates that gains arrive late.

Class-level AP averaged over the same IoU thresholds (AP@50:95) is used to determine whether aggregate improvement comes from lower-count classes or from already well-represented classes. Metal, Cardboard, and PETOil receive particular attention, but all eight classes are reported. After each acquisition, the number of ground-truth boxes added for each class is counted for analysis only. Comparing class AP with labeled instances separates exposure from learning response. More boxes with little AP gain suggest redundant, ambiguous, or poorly targeted supervision. An AP gain without a large count increase suggests higher-value examples.

For selection behavior and annotation efficiency, two additional audits are used. First, the budget curves are re-indexed by cumulative ground-truth box count. If an image-budget advantage becomes much smaller on the box axis, part of the apparent efficiency came from selecting denser images rather than reducing bounding-box work. Actual labor savings cannot be claimed without annotation-time measurements. Second, temporal concentration is the proportion of queried images that have another image from the same video in the same acquisition batch within five frame indices. A high short-range neighbor rate indicates that a query batch concentrates on closely related frames even when its feature-space diversity score is favorable.

Each metric is computed independently for each of the six seeds and summarized as mean ± sample standard deviation (SD). Paired differences from Random are calculated within seed, because all acquisition strategies share the same initialization for that seed. A high mean with a large standard deviation indicates sensitivity to the sampled initialization or stochastic training. The 95% intervals are unadjusted paired t-intervals over the six seeds, with seeds as the unit of replication. They quantify variation across seeds for single training runs and are not adjusted for multiple comparisons. An interval that excludes zero is reported as such, and an interval that includes zero is reported as inconclusive for the direction of the difference. One training run per seed and budget does not establish a fully replicated estimate or a definitive strategy ranking.

nAUBC and the 20% endpoint identify when an acquisition strategy performs well. Class-level AP and labeled instances identify which classes account for the difference, while box-budget and temporal analyses describe the selected data. The benchmark therefore assesses performance jointly in terms of detector improvement, annotation volume, and coverage of the video-derived pool.

B. Overall Detection Performance (RQ1)

1) Analysis scenario: For RQ1, each strategy is evaluated over the shared 5%–20% labeled-image trajectory for six paired seeds. The primary comparison is image-budget nAUBC, with test mAP@50:95 at 20% reported as the endpoint. Following area-under-budget evaluation in [15] and [16], nAUBC is computed independently for each seed before aggregation. All budget points are independently retrained from the same COCO-pretrained initialization, so differences are paired by seed and acquisition strategy.

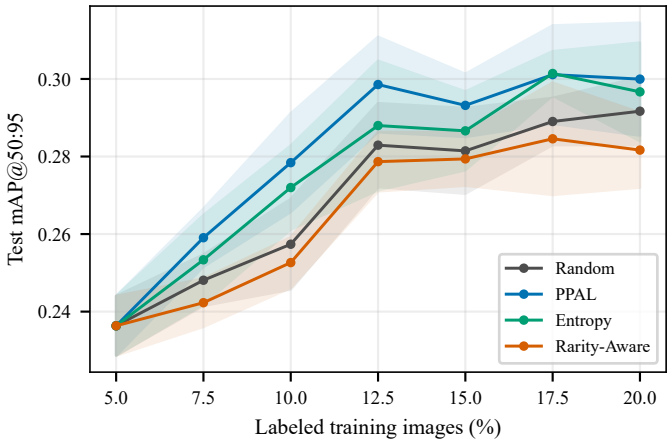

Fig. 1. Test mAP@50:95 across labeled-image budgets. Thick lines show means across six seeds; shaded bands show one sample standard deviation.

2) Results and discussion: PPAL and Entropy formed the leading group under the evaluated SortWaste–YOLOv8n protocol. PPAL achieved the highest mean nAUBC over the 5%–20% trajectory and the highest mean mAP@50:95 at 20%. Both summaries are reported to distinguish performance across acquisition rounds from the final operating point.

For seed s, let $n_{s,i}$ be the labeled-image count at budget point i, N_{train} the number of training images, $b_{s,i} = n_{s,i}/N_{\text{train}}$ the realized labeled fraction, and $m_{s,i}$ the corresponding test mAP@50:95. The seed-specific value is calculated using Eq. (13) [15], [16]:

$$\text{nAUBC}^{(s)}_{[0.05,0.20]} = \frac{\sum_{i=0}^{5} \frac{m_{s,i} + m_{s,i+1}}{2} (b_{s,i+1} - b_{s,i})}{b_{s,6} - b_{s,0}}. \quad (13)$$

The endpoint alone cannot describe performance before the 20% budget. Curve-level ranking used an area-under-the-budget construction established in active-learning evaluation [15], [16]. Dividing the trapezoidal area by the realized budget span expresses nAUBC as mean linearly interpolated test mAP@50:95 over the evaluated interval. It was calculated separately for each seed before reporting the mean and sample standard deviation across six paired seeds. Section IV-D2 examines the same curves against cumulative box count.

The mean margins of PPAL and Entropy over Random were largest at 10% and smaller at 20%. All four strategies started from the same 5% labeled set within each seed. Fig. 1 presents the budget trajectories, and Table II reports their mean values at each labeled-image budget.

At 10%, PPAL and Entropy exceeded Random by 0.0210 and 0.0146 mAP@50:95. Their margins declined to 0.0083 and 0.0050 at 20%. Rarity-Aware's mean remained below Random at every budget after the shared 5% point. The mean curves are consistent with an intermediate-budget advantage for PPAL

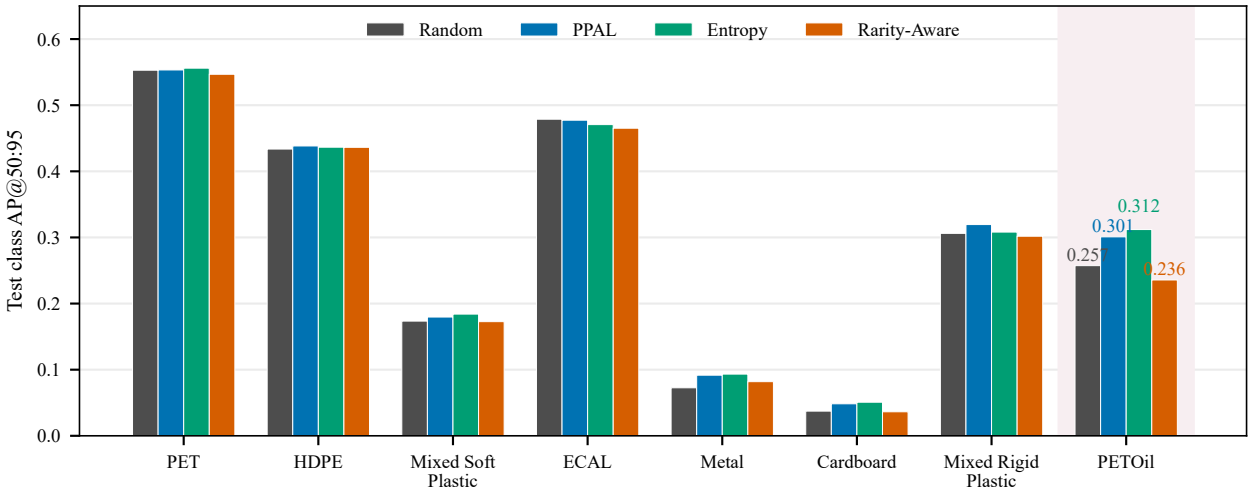

Fig. 2. Mean class AP@50:95 at 17.5% labeled images across six paired seeds. The shaded region highlights PETOil.

TABLE II. MEAN TEST MAP@50:95 ACROSS LABELED-IMAGE BUDGETS. VALUES ARE MEANS ACROSS SIX PAIRED SEEDS; BOLD IDENTIFIES THE HIGHEST MEAN AT EACH BUDGET

Labeled images	Random	PPAL	Entropy	Rarity-Aware
5%	0.2363	0.2363	0.2363	0.2363
7.5%	0.2481	**0.2591**	0.2534	0.2423
10%	0.2574	**0.2784**	0.2720	0.2526
12.5%	0.2829	**0.2986**	0.2880	0.2787
15%	0.2815	**0.2932**	0.2866	0.2794
17.5%	0.2890	0.3011	**0.3014**	0.2846
20%	0.2917	**0.3000**	0.2967	0.2817

and Entropy. Paired intervals were computed for nAUBC and for the 20% endpoint, not for each intermediate budget.

At 20%, the paired differences from Random were $+0.0083$ ($[-0.0086, 0.0252]$) for PPAL, $+0.0050$ ($[-0.0061, 0.0160]$) for Entropy, and -0.0100 ($[-0.0283, 0.0083]$) for Rarity-Aware. All three intervals include zero, so the endpoint does not separate any strategy from Random. The discussion below examines diagnostic evidence associated with the narrowing of the PPAL and Entropy margins as the labeled set grew.

PPAL ranked first by image-budget nAUBC, with Entropy close behind. PPAL exceeded Random in all six paired seeds, Entropy in five, and Rarity-Aware in one. Table III summarizes the curve-level rankings and paired differences.

Relative to Random, mean nAUBC was higher by 0.0126 for PPAL (unadjusted 95% paired t-interval [0.0032, 0.0220]) and by 0.0075 for Entropy ([0.0020, 0.0129]). Both intervals exclude zero. Rarity-Aware differed by -0.0044 ($[-0.0124, 0.0036]$), and its interval includes zero. The PPAL–Entropy difference was 0.0051 ($[-0.0053, 0.0155]$); its interval includes zero. These intervals are across six paired seeds, with one training run per seed and budget, and are not adjusted for multiple comparisons.

A cross-budget comparison is also available. With 463 labeled images at 12.5%, mean PPAL mAP@50:95 was 0.2986, compared with 0.2917 for Random at 20% with 741 images. This comparison uses means only; the associated box load is examined in Section IV-D2.

The curves were not monotonic. Across the 36 seed-level transitions for each strategy, declines occurred 7 times for Random, 8 for PPAL, 7 for Entropy, and 11 for Rarity-Aware. Each budget initiated an independent stochastic optimization run from the same pretrained weights. A local decline therefore cannot be assigned solely to the newly acquired images. nAUBC reduces the influence of any single upward or downward fluctuation by summarizing all seven budgets.

C. Lower-Count-Class Performance (RQ2)

1) Analysis scenario: For RQ2, AP@50:95, class nAUBC, annotation exposure, and alternative class scorings are analyzed across six paired seeds, with focused analysis of Metal, Cardboard, and PETOil. Ground-truth box counts are audited after acquisition and are unavailable to the acquisition strategies. Checkpoint-specific diagnostics are identified separately.

2) Results and discussion:

a) Class-level separation and the role of PETOil: SortWaste evaluates detection by macro-averaging AP over eight classes [1]. At any labeling budget, the macro average over class index c is defined by Eq. (14):

$$\mathrm{mAP}_{50:95} = \frac{1}{8}\sum_{c=1}^{8} \mathrm{AP}_{c,50:95}\,. \tag{14}$$

Each class consequently contributes one eighth of the reported mAP, regardless of how many test boxes it contains. PETOil has 132 test boxes and HDPE has 3,269, yet a change of 0.08 AP in either class changes mAP by the same 0.01 [1]. PETOil therefore carries equal metric weight with much smaller evaluation support. Fig. 2 compares class-level performance at 17.5%, the budget at which PPAL and Entropy reach their highest mean mAP@50:95 (Table II).

The six-seed analysis identifies PETOil as the largest individual contributor to the trajectory-level gain: it supplies

TABLE III. CURVE-LEVEL PERFORMANCE AND PAIRED DIFFERENCES FROM RANDOM. NAUBC VALUES ARE MEANS ACROSS SIX PAIRED SEEDS; BUDGET-POINT WINS EXCLUDE THE SHARED 5% INITIALIZATION POINT

Selection strategy	Image-budget nAUBC, mean ± sample SD	Paired nAUBC difference from Random, mean ± sample SD	Paired budget-point wins over Random
Random	0.2705 ± 0.0057	Reference	Reference
PPAL	**0.2831 ± 0.0079**	**+0.0126 ± 0.0090**	**30 of 36**
Entropy	0.2780 ± 0.0063	+0.0075 ± 0.0052	27 of 36
Rarity-Aware	0.2661 ± 0.0057	-0.0044 ± 0.0076	13 of 36

46% of the PPAL improvement and 53% of the Entropy improvement. At 17.5%, the corresponding shares are 45% and 55%. Substantial contributions also arise from other classes. Table IV reports positive mean per-class nAUBC differences for PPAL in all eight classes and for Entropy in seven. Metal, Mixed Rigid Plastic, and Cardboard contribute 18%, 14%, and 10% of the PPAL gain, respectively. Several low-AP classes therefore contribute substantially to the improvement, consistent with uncertainty- and difficulty-based acquisition.

Test predictions from the saved checkpoints were rescored over the seven classes excluding PETOil and with PETOil labels and predictions merged into PET, following the plastic-only SortWaste class grouping [1]. For the merged variant, NMS was repeated at IoU 0.7. With PETOil excluded, PPAL's paired nAUBC difference from Random was +0.0078 (unadjusted 95% paired t-interval [0.0026, 0.0131]). With PETOil merged into PET, the difference was +0.0080 ([0.0028, 0.0132]). Entropy's corresponding differences were +0.0040 and +0.0043; both intervals include zero (Table V). The PPAL PETOil AP margin over Random decreased from $+0.084$ at 10% to $+0.030$ at 20%. These values are differences of unrounded means (Fig. 4 and Table VI).

Table VI complements the trajectory analysis with class AP at the 20% endpoint.

At 20%, PPAL obtains the highest mean AP in PET, Mixed Soft Plastic, ECAL, Mixed Rigid Plastic, and PETOil, whereas Entropy leads in Metal and Cardboard and Random in HDPE. The six-seed endpoint means are therefore consistent with the aggregate PPAL ordering, with distinct class-specific leaders across strategies.

Table VII reports class-level precision and recall at 20%; boldface marks the maximum of each metric separately. PPAL has the highest mean PETOil precision, while Random has slightly higher PETOil recall. Entropy records competitive recall for several lower-count classes, and Rarity-Aware does not lead consistently across classes. Fig. 3 reports where ground-truth boxes were assigned. Across the four strategies, 0.11–0.20 of PETOil boxes were assigned to PET, 0.13–0.21 to HDPE, and 0.11–0.21 to Mixed Rigid Plastic, and 0.07–0.11 were missed. The fraction assigned to PETOil was 0.38 for Random, 0.38 for PPAL, 0.35 for Entropy, and 0.25 for Rarity-Aware. Metal boxes were most often assigned to ECAL (0.34–0.39) or missed (0.30–0.32), and 0.06–0.08 were assigned to Metal. Cardboard boxes were most often missed (0.43–0.49), and 0.12–0.16 were assigned to Cardboard. Table VII and Fig. 3 use different operating points: the table uses the maximum-F1 confidence at IoU 0.50, and the figure uses confidence 0.25 at IoU 0.45. Precision is normalized by predictions, recall by ground-truth boxes, and each figure row by ground-truth boxes, so the values are not interchangeable. In a sorting line, missed detections can leave objects unsorted, class confusions can route materials incorrectly, localization errors can shift a grasp point, and duplicate detections can prompt repeated handling.

AP@50:95 evaluates detection across multiple IoU thresholds, but the evaluation outputs do not separately enumerate duplicate detections or localization errors. Error decomposition frameworks such as TIDE distinguish these contributions [17]; that decomposition was not performed. The AP curves and confusion matrices summarize detection quality and class assignments, while the PET–PETOil example illustrates class ambiguity.

b) The acquisition strategies targeted different bottlenecks in sortwaste: The acquisition strategies differ along two axes that matter for SortWaste: how they estimate unresolved detector error and whether a candidate's value changes after other images enter the same batch. PPAL combines training-derived classification and localization difficulty with joint CCMS selection. Entropy combines predictive entropy with ENMS and accepted-set prototype updates. Rarity-Aware combines predicted scarcity with class-conditioned quotas, while its image scores remain fixed during batch construction. Table VIII summarizes these differences in acquisition signals, redundancy control, and class allocation.

PPAL's training-derived difficulty signal provides a mech-

TABLE IV. CLASS NAUBC OVER 5%–20% LABELED IMAGES ACROSS SIX PAIRED SEEDS. RANDOM REPORTS MEAN CLASS NAUBC; OTHER COLUMNS REPORT MEAN DIFFERENCES FROM RANDOM. PARENTHESES COUNT SEEDS WITH POSITIVE DIFFERENCES

Class	Random	PPAL	Entropy	Rarity-Aware
PET	0.531	+0.0048 (4)	+0.0051 (4)	-0.0045 (1)
ECAL	0.461	+0.0008 (4)	-0.0027 (3)	-0.0095 (1)
HDPE	0.426	+0.0027 (4)	+0.0028 (3)	-0.0047 (2)
Mixed Rigid Plastic	0.280	+0.0146 (6)	+0.0009 (3)	-0.0065 (1)
PETOil	0.183	+0.0459 (5)	+0.0318 (5)	-0.0166 (1)
Mixed Soft Plastic	0.171	+0.0041 (4)	+0.0033 (3)	-0.0076 (1)
Metal	0.073	+0.0179 (4)	+0.0146 (4)	+0.0106 (5)
Cardboard	0.039	+0.0100 (4)	+0.0041 (4)	+0.0037 (3)

TABLE V. PAIRED NAUBC DIFFERENCES FROM RANDOM UNDER ALTERNATIVE CLASS SCORINGS ACROSS SIX SEEDS. BRACKETS GIVE UNADJUSTED 95% PAIRED T-INTERVALS; PARENTHESES COUNT SEEDS WITH POSITIVE DIFFERENCES. PETOIL SHARE DENOTES ITS CONTRIBUTION TO THE EIGHT-CLASS MACRO DIFFERENCE

Strategy	Eight classes	Without PETOil	PETOil merged into PET	PETOil share
PPAL	$+0.0126$ $[0.0032, 0.0220]$ (6)	$+0.0078$ $[0.0026, 0.0131]$ (6)	$+0.0080$ $[0.0028, 0.0132]$ (6)	46% of gain
Entropy	$+0.0075$ $[0.0020, 0.0129]$ (5)	$+0.0040$ $[-0.0006, 0.0086]$ (5)	$+0.0043$ $[-0.0001, 0.0088]$ (5)	53% of gain
Rarity-Aware	-0.0044 $[-0.0124, 0.0036]$ (1)	-0.0026 $[-0.0067, 0.0014]$ (2)	-0.0025 $[-0.0066, 0.0017]$ (2)	47% of deficit

TABLE VI. CLASS AP@50:95 AT 20% LABELED IMAGES ACROSS SIX PAIRED SEEDS. BOLD IDENTIFIES THE HIGHEST MEAN FOR EACH CLASS

Class	Random	PPAL	Entropy	Rarity-Aware
PET	0.552	**0.558**	0.556	0.547
HDPE	**0.438**	0.431	0.431	0.431
Mixed Soft Plastic	0.174	**0.182**	0.170	0.161
ECAL	0.470	**0.477**	0.471	0.466
Metal	0.072	0.078	**0.092**	0.091
Cardboard	0.028	0.041	**0.047**	0.033
Mixed Rigid Plastic	0.308	**0.312**	0.305	0.295
PETOil	0.290	**0.321**	0.301	0.229

TABLE VII. PER-CLASS PRECISION AND RECALL AT THE 20% LABELED-IMAGE BUDGET. EACH CELL REPORTS MEAN PRECISION/RECALL ACROSS THE SIX SEEDS. VALUES ARE ULTRALYTICS 8.4.51 PER-CLASS RESULTS AT IOU 0.50 AND AT THE CONFIDENCE THAT MAXIMIZES THE SMOOTHED MEAN F1 OVER CLASSES

Class	Random	PPAL	Entropy	Rarity-Aware
PET	0.724/0.734	**0.750/0.736**	0.736/0.729	0.709/0.728
HDPE	**0.621/0.536**	0.614/0.516	0.593/0.535	0.608/0.529
Mixed Soft Plastic	0.348/**0.334**	**0.383**/0.309	0.361/0.323	0.345/0.317
ECAL	0.581/**0.693**	**0.606**/0.666	0.578/0.669	0.572/0.680
Metal	0.246/0.095	0.236/0.070	**0.294**/0.102	0.267/**0.104**
Cardboard	0.085/0.116	0.109/0.137	**0.122/0.159**	0.100/0.135
Mixed Rigid Plastic	**0.407**/0.474	0.397/0.491	0.395/**0.496**	0.381/0.495
PETOil	0.473/**0.441**	**0.525**/0.439	0.516/0.398	0.445/0.329

TABLE VIII. ACQUISITION SIGNALS, WITHIN-BATCH REDUNDANCY CONTROL, AND CLASS ALLOCATION. THE TABLE SUMMARIZES THE SOURCE METHODS AND THEIR YOLOV8 ADAPTATIONS USED IN THIS BENCHMARK [8], [9], [10]

Strategy	Acquisition mechanism
PPAL	*Signal:* Training-derived classification and localization difficulty applied to uncertainty. *Redundancy control:* CCMS compares multi-object candidates, followed by representative selection. *Class allocation:* Difficulty reweighting without a fixed class quota.
Entropy	*Signal:* Binary prediction entropy refined by ENMS. *Redundancy control:* DivProto compares class-specific prototypes with images already selected. *Class allocation:* Explicit allocation toward minority classes.
Rarity-Aware	*Signal:* Rarity-weighted uncertainty, confidence, and fixed diversity terms. *Redundancy control:* Embedding distance uses fixed pool centroids and ignores partial-batch selections. *Class allocation:* Rarity quotas followed by global score ranking.

anism for weighting unresolved classification and localization errors. Following Yang et al. [8], for each positive training assignment indexed by j, the YOLOv8 adapter computes assignment quality q_j and the difficulty d_c of class c, as shown in Eq. (15):

$$q_j = p_{j,\mathrm{gt}}^{0.6}\,\mathrm{IoU}(b_j, \hat{b}_j)^{0.4}, \qquad d_c = 1 - Q_c. \tag{15}$$

Here, the probability term denotes confidence in the assigned ground-truth class, and the two boxes denote the prediction and its assigned ground-truth box. The class-quality statistic is the exponential moving average of assignment quality, and class difficulty reweights instance uncertainty. By construction, DCUS combines classification confidence and localization quality [8]; it is not an inverse-frequency or

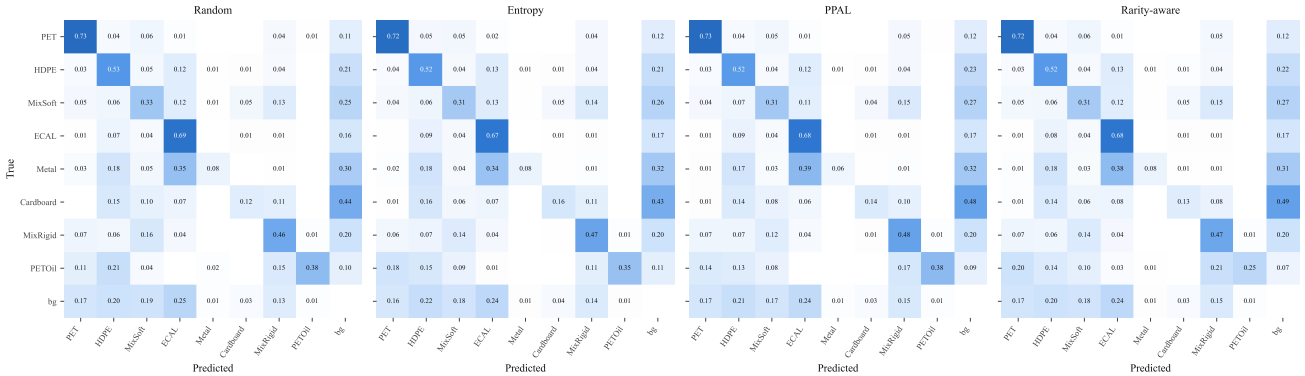

Fig. 3. Normalized test-set confusion matrices at the 20% labeled-image budget. Each matrix aggregates the six seeds for one acquisition strategy. Rows denote ground-truth classes and columns denote predicted classes; values are normalized within each row. The background column summarizes missed detections, and the background row summarizes false-positive assignments. Matching uses confidence above 0.25 and IoU above 0.45, the Ultralytics 8.4.51 defaults for confusion matrices. Cells with values that round to 0.00 are left unlabeled.

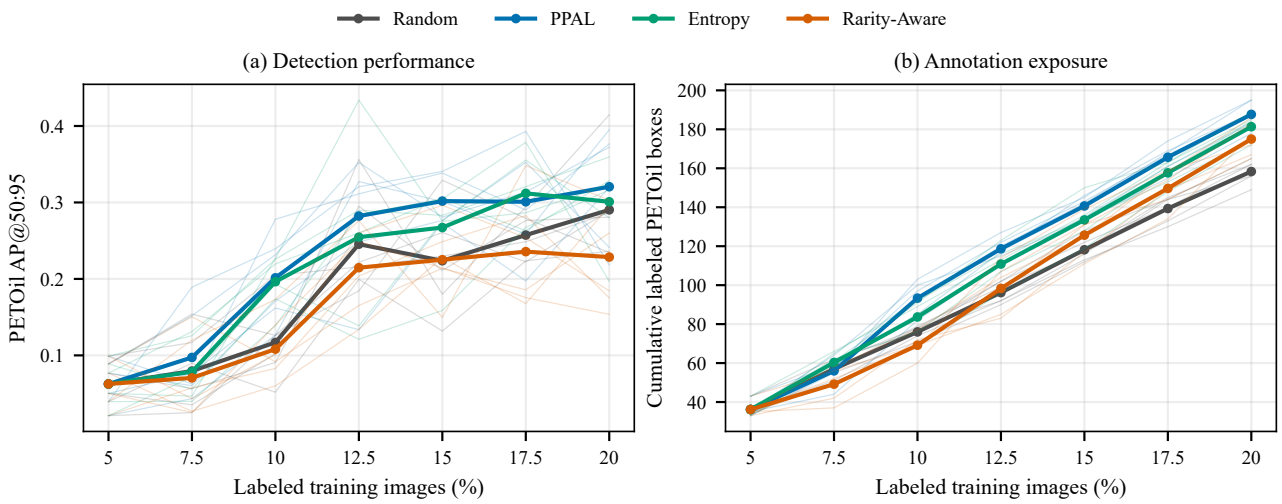

Fig. 4. PETOil AP@50:95 and cumulative labeled boxes across six paired seeds. Thin lines show individual seeds; thick lines show means. Box counts were audited after acquisition and were unavailable to acquisition strategies.

inverse-AP score. This mechanism provides context for the class-level results without separating its contribution from the other PPAL components.

PPAL's second stage addresses coverage within the difficult candidate pool. The YOLOv8 adapter matches detections between image pairs, compares normalized class-score vectors through symmetrized directional KL distance, and selects representatives jointly. This selection operates in class-posterior space and does not explicitly constrain frame distance. The six-seed temporal analysis in RQ3 therefore describes a separate property of the selected batches.

Entropy combines predictive entropy with progressive diversity: ENMS suppresses redundant same-class detections, and DivProto updates class prototypes as images enter the batch [10]. The source method also assigns at least 75% of the budget to the least represented half of the classes.

Fig. 4 relates six-seed PETOil accuracy to annotation exposure. At 17.5%, Entropy and Rarity-Aware had accumulated 157.7 and 149.7 PETOil boxes on average, with AP values of 0.312 and 0.236. PPAL reached 0.301 with 165.7 boxes. These differences are consistent with class exposure alone not explaining the class response; the selected examples and detector training may also contribute.

Fig. 5 presents the visual ambiguity behind the class-level result. The object retains the shape and transparency of ordinary PET while showing oil contamination and deformation. SortWaste defines PETOil as a PET subcategory and merges it into PET in the plastic-only benchmark because the two classes share material properties and visual appearance [1]. The independently trained models changed from a PET error at 5% to a weak PETOil prediction at 17.5%. This example is consistent with the PET–PETOil confusion identified by the quantitative analysis, but it is a best-case illustration rather than representative evidence by itself. The aggregate class conclusions use the six-seed analysis across the complete 5%–20% budget trajectory. Per-image prediction files from the evaluation passes were not archived, so persistent failures and newly introduced errors could not be enumerated separately.

The published full-supervision result provides additional context: YOLOv11 reached 0.451 mAP@50:95 overall and 0.725 AP@50:95 on PETOil [1]. Differences in detector and training protocol preclude a direct annotation-budget comparison.

c) Class coverage and detection response of the rarity-aware adaptation: The six-seed exposure audit (Table IX) reports that Rarity-Aware achieved class coverage comparable to PPAL and Entropy, with the highest Metal exposure. The mean Metal nAUBC difference from Random was +0.0106, with positive differences in five of six seeds; the corresponding Cardboard values were +0.0037 and three of six. However, PPAL and Entropy achieved larger mean Metal gains with

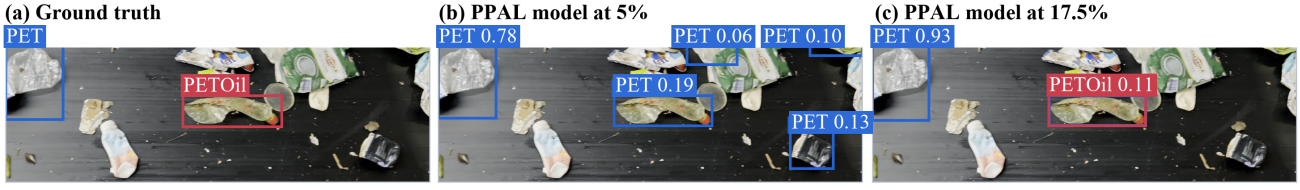

Fig. 5. Best-case qualitative PETOil example from test frame 0008_2564_5 for seed 13. Among screened PETOil-positive images, this frame showed the largest spatially matched improvement: the 5% model predicts PET, whereas the independently trained 17.5% model predicts PETOil. Predictions are shown at confidence 0.05. This example is not representative of typical detector behavior.

TABLE IX. SIX-SEED CLASS EXPOSURE AMONG THE 556 IMAGES ACQUIRED FROM 5% TO 20%. COVERAGE IS THE PERCENTAGE CONTAINING METAL, CARDBOARD, OR PETOIL; BOX COUNTS ARE MEAN QUERIED BOXES PER SEED

Strategy	Coverage	Metal boxes	PETOil boxes
Random	58.2%	143.2	122.2
PPAL	68.1%	203.5	151.5
Entropy	68.9%	207.5	145.2
Rarity-Aware	69.6%	238.2	138.8

fewer queried Metal boxes. Rarity-Aware's macro difference of -0.0044 comprises -0.0041 from the five higher-count classes and -0.0021 from PETOil, partly offset by +0.0018 from Metal and Cardboard. These results describe the class-specific strengths and trade-offs of the evaluated adaptation.

The approximately 21-fold instance imbalance overstated encounter scarcity at image level. Metal, Cardboard, and PETOil appeared in 20.9%, 33.3%, and 19.0% of training images, so Random encountered each class regularly. Access to PETOil was only part of the problem. A possible explanation for the larger separation is that the selected examples resolved more of the remaining PET–PETOil error; this was not tested.

The routing audit identified a first constraint in the evaluated configuration. Rarity-Aware estimates class weights from predicted detection frequency, combines them with labeled class balance, and reserves part of the query for class-conditioned allocation [9]. Its source paper likewise recognizes early prediction bias as a limitation. In the audited initial pool for seed 1337, the YOLOv8 Top-300 adapter retained 300 detections for every image. It predicted PETOil in 3,493 of 3,520 images, while revealed labels placed PETOil in 672. The resulting 19.2% precision left the presence-based fallback with little power to distinguish relevant images. Routing first considers the dominant predicted class, so the 19.2% value applies only to the fallback stage. Raising confidence to 0.05 changed 24 of 93 selections at the same checkpoint. This diagnostic indicates sensitivity of the selected batch to detector-specific retention, but it does not establish a training or AP effect.

Batch construction creates a second structural difference. Rarity-Aware computes a composite score for every pool image before constructing the query, directs half of the budget through class-conditioned quotas, and fills the remaining positions by score rank. Accepting an image does not update the scores of the remaining candidates, so a similar candidate retains its original value. Following Benhafid and Selouani [9], its embedding term E_i for image i is defined in Eq. (16):

$$E_i = \min_{\mu \in \mathcal{M}} \|z_i - \mu\|_2, \tag{16}$$

where, z_i is the ℓ_2-normalized embedding of image i, $\mathcal{M}$ is the set of unlabeled-pool cluster centroids, and $\mu \in \mathcal{M}$ denotes a candidate centroid. Since E_i is measured against fixed centroids, it remains unchanged as the batch grows. Entropy updates DivProto prototypes from accepted images, while PPAL performs joint medoid selection over its pairwise CCMS matrix. A possible explanation is that the weak fallback signal and scores that do not react to accepted images allowed correlated frames to retain high ranks. The six-seed temporal audit in Section IV-D provides related evidence. Its effect on AP can be tested through a full acquisition-and-training ablation.

The source Rarity-Aware experiments used YOLOv10 and YOLOv12 with warm continuation [9]. The present comparison evaluates transfer to YOLOv8n under shared independent retraining. The routing probe identifies weak PETOil presence discrimination in one checkpoint; its contribution to AP and the losses on higher-count classes require separate acquisition-and-training ablations.

D. Selection Behavior and Annotation Efficiency (RQ3)

1) Analysis scenario: For RQ3, cumulative labeled-box counts and short-range temporal concentration are audited across six paired seeds. Box-budget nAUBC uses linearly interpolated curves over the common observed box-count interval within each seed. Temporal concentration is the proportion of queried images with another image from the same video in the same acquisition batch within five frame indices, averaged across the six seeds and six acquisitions.

2) Results and discussion:

a) Annotation volume and temporal concentration: Equal labeled-image budgets did not produce equal annotation volumes. Across six seeds, PPAL, Entropy, and Rarity-Aware queried 20.56–20.88 ground-truth boxes per image, compared with 16.62 under Random. Table X combines annotation volume, box-budget performance, and temporal concentration. Counts were audited after acquisition and were unavailable to the acquisition strategies.

Within each acquisition batch, the proportions of images with a same-video neighbor within five indices are 21.0% for Random, 53.5% for PPAL, 63.5% for Entropy, and 69.1% for Rarity-Aware. These are means across 36 seed–round batches per strategy. Active acquisition strategies therefore have approximately 2.5–3.3 times Random's short-range neighbor rate. This within-batch measure is distinct from proximity between labeled and test frames.

b) Temporal-proximity sensitivity analysis: The official split assigns whole scenes to each subset and contains frames from two video sequences. To examine whether short-range proximity could explain the strategy differences, a post-hoc filtered-test analysis was performed. Frames were treated as close when their same-video index difference was at most 10 or 20 frames, corresponding to approximately 2 or 4 seconds at the dataset's 5 frames per second. The training pool contains 3,705 images and the test set contains 776 images. Under the scene-based split, the nearest training frame was at least six indices away for every test image, consistent with the five frames discarded at the start of each scene. Thirty images (3.9%) were within 10 indices of the training pool, and 90 images (11.6%) were within 20 indices.

At the 20% budget, the proportion of test images with a labeled image within 10 indices was 1.74% for Random, 2.92% for PPAL, 2.73% for Entropy, and 2.56% for Rarity-Aware. These proportions are small relative to the full test set, although the active acquisition strategies had higher proximity than Random. The same trained models were then re-evaluated

TABLE X. ANNOTATION VOLUME, BOX-BUDGET PERFORMANCE, AND WITHIN-BATCH TEMPORAL CONCENTRATION ACROSS SIX PAIRED SEEDS. A TEMPORAL NEIGHBOR IS ANOTHER QUERIED IMAGE FROM THE SAME VIDEO WITHIN FIVE FRAME INDICES. BOX-BUDGET DIFFERENCES USE COMMON OBSERVED BOX-COUNT SUPPORT WITHIN SEED

Selection strategy	Ground-truth boxes per queried image	Box-budget nAUBC difference from Random	Images with a same-video neighbor within five frame indices
Random	16.62	Reference	21.0%
PPAL	20.56	+0.00753	53.5%
Entropy	20.88	+0.00109	63.5%
Rarity-Aware	20.87	-0.01038	69.1%

after removing test images within 10 or 20 indices of the training pool. This procedure tests sensitivity to the affected test subset; it does not constitute a held-out-video split or a causal decomposition.

Filtering test images within 20 frame indices reduced the PPAL and Entropy differences by approximately 20% and 47%, respectively (Table XI). PPAL's interval remained above zero; Entropy's included zero. Rarity-Aware's difference became slightly more negative. These values describe sensitivity to the filtered subset, not the fraction of any gain attributable to temporal proximity: no randomly filtered control set was evaluated. A held-out-video comparison would require reserving one of the two videos for testing and retraining every strategy on the other.

c) Selection behavior of the rarity-aware top-300 adaptation: Across 36 acquisitions over six seeds, Rarity-Aware queried 833 PETOil boxes and Entropy 871; their PETOil nAUBC differed by 0.0484 in favor of Entropy. Separately, the checkpoint-specific retention probe for seed 1337 (Section IV-C) identifies a detector-transfer condition: predicted class presence must distinguish relevant images from the rest of the pool. Its effect on AP was not evaluated through retraining.

d) Detection-level score aggregation and annotation density: Detection-level score aggregation provides a possible explanation for the observed annotation density. PPAL sums difficulty-weighted entropy over retained detections, Entropy sums entropy after ENMS, and Rarity-Aware includes detection-level uncertainty and information sums. Object-dense images therefore have more opportunities to accumulate score.

Across six seeds, box-based indexing reduced the PPAL difference from +0.0126 to +0.0075 nAUBC (95% interval $[-0.0018, 0.0168]$) and the Entropy difference from +0.0075 to +0.0011 ($[-0.0053, 0.0075]$). Both intervals include zero. The image- and box-budget analyses distinguish gains under a fixed image count from gains under comparable bounding-box volume. The smaller differences are consistent with a ranking that depends in part on the budget definition, particularly for Entropy. Rarity-Aware's difference changed from -0.0044 to -0.0104 ($[-0.0167, -0.0041]$), and this unadjusted interval excludes zero.

The acquisition strategies do not encode video identity or frame distance, so neighboring frames remain eligible when their model-derived signals differ. Image count, object count, and temporal coverage therefore describe separate properties of the acquired data.

E. Integrated Discussion

1) Cross-RQ synthesis: The class analysis identifies gains beyond one category. PETOil is the largest individual contributor to the PPAL and Entropy improvements; Metal, Mixed Rigid Plastic, and Cardboard also contribute. Rarity-Aware increases lower-count-class exposure, but its mean image-budget nAUBC (0.2661) is below Random (0.2705). Class exposure and aggregate detection performance therefore describe distinct outcomes of acquisition.

2) Narrowing margins at larger budgets: Random's cumulative PETOil exposure increased from 36.2 to 158.3 labeled boxes between 5% and 20%. Its within-batch five-frame neighbor rate was 21.0%, compared with 53.5%–69.1% for the active strategies. These selection patterns accompany the narrowing mean performance margins.

At 20%, the mean mAP@50:95 margins were +0.0083 for PPAL and +0.0050 for Entropy, and both intervals include zero (Section IV-B). PPAL's PETOil AP margin was +0.030. The smaller endpoint margins coincide with increased cumulative coverage under Random.

The PPAL source study reported larger early-cycle gains that narrowed later [8]. A similar timing appears on SortWaste: PPAL and Entropy had their largest aggregate margins at 10%, with positive mean differences at 20%.

3) Relation to source studies and transfer limits: The source studies report AP or mAP at successive budgets and differ in detector and cross-cycle training history. PPAL's official implementation restarts detector training at each budget [18], matching the independent YOLOv8n runs used here. Rarity-Aware initializes each cycle from the previous best validation checkpoint [9]; the Entropy source study does not specify cross-cycle weight transfer or checkpoint selection [10]. These differences preclude direct comparison of absolute scores.

The PPAL study reported a 6.0-point mAP@50 advantage over Random on the PASCAL Visual Object Classes (VOC) dataset at 20% [8]. The present margin is $+0.0083$ mAP@50:95, which equals 0.83 points on a 0–100 scale. The two values use different metrics (mAP@50 versus mAP@50:95), datasets, and detectors, so they are not directly comparable. Entropy's positive mean difference is directionally consistent with reported gains on COCO and YOLOv5 [10]. Rarity-Aware was evaluated in its source study with YOLOv10 and YOLOv12 and warm continuation to 15% [9]; the current results describe its transfer to independently trained YOLOv8n.

The confusion patterns in Section IV-C have operational implications. The consequences of missed detections, class

TABLE XI. SENSITIVITY OF PAIRED NAUBC DIFFERENCES FROM RANDOM TO FILTERED TEST SETS. VALUES ARE MEAN DIFFERENCES ACROSS SIX PAIRED SEEDS; BRACKETS SHOW UNADJUSTED 95% PAIRED T-INTERVALS

Strategy	All test images	After removing ≤ 10 frames	After removing ≤ 20 frames	Relative reduction at 20 frames
PPAL	$+0.0126$ $[0.0032, 0.0220]$	$+0.0101$ $[0.0012, 0.0190]$	$+0.0101$ $[0.0011, 0.0191]$	20%
Entropy	$+0.0075$ $[0.0020, 0.0129]$	$+0.0048$ $[-0.0011, 0.0106]$	$+0.0040$ $[-0.0027, 0.0107]$	47%
Rarity-Aware	-0.0044 $[-0.0124, 0.0036]$	-0.0065 $[-0.0137, 0.0007]$	-0.0074 $[-0.0150, 0.0002]$	—

confusions, localization errors, and duplicate detections for sorting are described there.

F. Limitations

Each seed–budget point was trained once. Six paired seeds quantify variation across these runs but do not separate stochastic training effects from selected-set effects. Paired t-intervals are not adjusted for multiple comparisons and treat seeds as replicates.

SortWaste contains two video sequences. Filtering test images within 20 frame indices evaluates sensitivity to short-range proximity; longer-range similarity and transfer to unseen videos were not evaluated. The results are specific to SortWaste, YOLOv8n, six seeds, and the 5%–20% labeled-image-budget range.

Rarity-Aware results concern the YOLOv8 Top-300 adaptation with independent retraining. Higher-confidence retention variants and warm continuation were not evaluated. Source-based method parameters and detector-adapter settings were fixed before evaluation, with no strategy-specific tuning. Native-configuration comparisons and component ablations remain untested.

The benchmark compares detection effectiveness under fixed image budgets; computational cost was not a primary outcome. Per-round acquisition latency, candidate scoring, feature extraction, and peak memory were not separately profiled; recorded round time combines detector training and evaluation. Annotation drawing and verification time were not recorded, so box counts measure annotation volume rather than labor. End-to-end sorting performance and target-hardware throughput were not evaluated.

V. CONCLUSION

The six-seed benchmark provides a controlled comparison of four acquisition strategies for class-imbalanced waste detection on SortWaste. Under the shared YOLOv8n protocol, PPAL had the highest mean nAUBC (0.2831 ± 0.0079), followed by Entropy (0.2780 ± 0.0063), Random (0.2705 ± 0.0057), and Rarity-Aware (0.2661 ± 0.0057). Mean paired nAUBC differences from Random were positive for PPAL and Entropy, with unadjusted intervals that excluded zero, while the PPAL–Entropy interval included zero. At 20%, the paired endpoint differences from Random had intervals that included zero for all three non-random strategies.

PETOil was the largest individual class contributor to the PPAL and Entropy gains; Metal, Mixed Rigid Plastic, and Cardboard also contributed. PPAL's paired nAUBC differences from Random were +0.0078 with PETOil excluded and +0.0080 with PETOil merged into PET. The evaluated Rarity-Aware adaptation increased lower-count-class exposure, while its aggregate nAUBC remained below Random.

Across six seeds, active strategies selected 20.56–20.88 ground-truth boxes per queried image, compared with 16.62 for Random. Their within-batch five-frame neighbor rates were 53.5%–69.1%, versus 21.0% for Random. Under the matched box-count budget, mean nAUBC differences from Random were +0.0075 for PPAL (unadjusted 95% paired t-interval $[-0.0018, 0.0168]$) and +0.0011 for Entropy ($[-0.0053, 0.0075]$); both intervals included zero. The Rarity-Aware difference was -0.0104 ($[-0.0167, -0.0041]$). Image-budget gains coincided with higher annotation volume and short-range temporal concentration, while box-budget differences were smaller and uncertain. These findings apply to SortWaste, YOLOv8n, six seeds, and the 5%–20% labeled-image range.

DECLARATIONS

Declaration on Generative AI

During the preparation of this manuscript, the authors used OpenAI ChatGPT/Codex to assist with language refinement, LaTeX formatting, code review, and consistency checking. All AI-assisted outputs were critically reviewed and revised by the authors. The authors take full responsibility for the scientific content, experimental results, interpretations, originality, and integrity of the manuscript.

Code Availability

The source code for this study is available at https://github.com/chunhi-Mou/controlled-evaluation-of-active-learning-strategies.

REFERENCES

[1] S. Inácio, H. Proença, and J. C. Neves, "SortWaste: A densely annotated dataset for object detection in industrial waste sorting," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. Workshops (WACVW)*, Tucson, AZ, USA, Mar. 2026, pp. 576–584. [Online]. Available: https://openaccess.thecvf.com/content/WACV2026W/WasteVision/html/Inacio_SortWaste_A_Densely_Annotated_Dataset_for_Object_Detection_in_Industrial_WACVW_2026_paper.html

[2] G. Jocher, A. Chaurasia, and J. Qiu, *Ultralytics YOLO*, version 8.4.51, computer software, PyPI, May 2026. Accessed: Aug. 24, 2026. [Online]. Available: https://pypi.org/project/ultralytics/8.4.51/

[3] J. M. Johnson and T. M. Khoshgoftaar, "Survey on deep learning with class imbalance," *J. Big Data*, vol. 6, no. 1, Art. no. 27, Mar. 2019, doi: https://doi.org/10.1186/s40537-019-0192-5.

[4] B. Settles, "Active learning literature survey," Dept. Comput. Sci., Univ. Wisconsin–Madison, Madison, WI, USA, Tech. Rep. 1648, 2009. Accessed: Aug. 24, 2026. [Online]. Available: https://hdl.handle.net/1793/60660

[5] Z. Feng, S. Zhang, R. Takezoe, W. Hu, M. Chandraker, L.-J. Li, V. K. Narayanan, and X. Wang, "ALBench: A framework for evaluating active learning in object detection," in *Proc. AutoML-Conf Late-Breaking Workshop*, 2022, arXiv:2207.13339, doi: https://doi.org/10.48550/arXiv.2207.13339.

[6] P. Munjal, N. Hayat, M. Hayat, J. Sourati, and S. Khan, "Towards robust and reproducible active learning using neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New Orleans, LA, USA, Jun. 2022, pp. 223–232, doi: https://doi.org/10.1109/CVPR52688.2022.00032.

[7] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Computer Vision–ECCV 2014*, vol. 8693, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham, Switzerland: Springer, 2014, pp. 740–755, doi: https://doi.org/10.1007/978-3-319-10602-1_48.

[8] C. Yang, L. Huang, and E. J. Crowley, "Plug and play active learning for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2024, pp. 17784–17793, doi: https://doi.org/10.1109/CVPR52733.2024.01684.

[9] Z. Benhafid and S. A. Selouani, "Rarity-aware stratified active learning for class-imbalanced industrial object detection," *Appl. Sci.*, vol. 16, no. 3, Art. no. 1236, Jan. 2026, doi: https://doi.org/10.3390/app16031236.

[10] J. Wu, J. Chen, and D. Huang, "Entropy-based active learning for object detection with progressive diversity constraint," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New Orleans, LA, USA, Jun. 2022, pp. 9387–9396, doi: https://doi.org/10.1109/CVPR52688.2022.00918.

[11] D. Yoo and I. S. Kweon, "Learning loss for active learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 93–102, doi: https://doi.org/10.1109/CVPR.2019.00018.

[12] C.-C. Kao, T.-Y. Lee, P. Sen, and M.-Y. Liu, "Localization-aware active learning for object detection," in *Computer Vision–ACCV 2018*. Cham, Switzerland: Springer, 2019, pp. 506–522, doi: https://doi.org/10.1007/978-3-030-20876-9_32.

[13] J. Choi, I. Elezi, H.-J. Lee, C. Farabet, and J. M. Alvarez, "Active learning for deep object detection via probabilistic modeling," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Montreal, QC, Canada, Oct. 2021, pp. 10244–10253, doi: https://doi.org/10.1109/ICCV48922.2021.01010.

[14] W. Yu, S. Zhu, T. Yang, and C. Chen, "Consistency-based active learning for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, New Orleans, LA, USA, Jun. 2022, pp. 3950–3959, doi: https://doi.org/10.1109/CVPRW56347.2022.00440.

[15] A. Pardo, M. Xu, A. Thabet, P. Arbeláez, and B. Ghanem, "BAOD: Budget-aware object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Nashville, TN, USA, Jun. 2021, pp. 1247–1256, doi: https://doi.org/10.1109/CVPRW53098.2021.00137.

[16] X. Zhan, H. Liu, Q. Li, and A. B. Chan, "A comparative survey: Benchmarking for pool-based active learning," in *Proc. 30th Int. Joint Conf. Artif. Intell. (IJCAI)*, Montreal, QC, Canada, Aug. 2021, pp. 4679–4686, doi: https://doi.org/10.24963/ijcai.2021/634.

[17] D. Bolya, S. Foley, J. Hays, and J. Hoffman, "TIDE: A general toolbox for identifying object detection errors," in *Computer Vision–ECCV 2020*. Cham, Switzerland: Springer, 2020, pp. 558–573, doi: https://doi.org/10.1007/978-3-030-58580-8_33.

[18] C. Yang, *PPAL: Plug and Play Active Learning for Object Detection*, official software repository, GitHub, 2024. Associated paper: C. Yang, L. Huang, and E. J. Crowley, "Plug and Play Active Learning for Object Detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2024, pp. 17784–17793, doi: https://doi.org/10.1109/CVPR52733.2024.01684. Accessed: Aug. 24, 2026. [Online]. Available: https://github.com/ChenhongyiYang/PPAL