Gradient-Guided Robust Adaptive Multiclass Object Detection in Sputum Cell Images

Yushu Liu[1], Wenhao Yi[1], Yuhua Feng[1]*, Ming Zhang[1],
Di Gai[1], Weiping Zhang[1], Xianhao Zhong[2]
[1] School of Mathematics and Computer Sciences,
Nanchang University, Nanchang 330029, China
[2] Faculty of Computer Science and Information Technology,
University of Malaya, Kuala Lumpur 50603, Malaysia

ABSTRACT
Automated sputum cell detection supports quantitative assessment of respiratory inflammation, but clinical images contain overlapping targets, heterogeneous backgrounds, large scale variation, and imbalanced classes. This study presents GRA-InternRTM, a multiclass detector based on InternImage-RTMDet. An Edge-Guided Deformable Adapter (EGDA) strengthens boundary representations, and an Adaptive Gated Fusion Module (AGFM) provides sample-specific multiscale fusion. Dynamic Head (DyHead) aligns features across scales, while Dual-Robust Learning loss (DRL loss) reduces the effects of class imbalance and missing annotations. Ablation experiments evaluated the four components on an in-house dataset. Compared with six reference detectors, GRA-InternRTM achieved mAP, mAP50, Recall, and AccMatched values of 0.365, 0.622, 0.8762, and 0.9617, respectively. Relative to Fast R-CNN, it improved mAP, mAP50, and AccMatched by 0.014, 0.024, and 0.0308. These results support the technical feasibility of multiclass detection in complex sputum cell images, although external validation remains necessary.

General Terms
Algorithms, Experimentation, Performance

Keywords
Sputum cell images, object detection, InternImage, EGDA, AGFM, DyHead

1. INTRODUCTION
Sputum cytology provides a minimally invasive and repeatable approach to characterizing airway inflammation, and integrating artificial intelligence with healthcare is encouraged by national policy [1]. Lung cancer control remains challenging [2], and sputum properties relate to disease severity in pneumonia and asthma phenotypes [3,4]. The images analyzed here contain six classes: neutrophils (NEU), macrophages (Mac), eosinophils (Eos), epithelial cells, bacteria, and lymphocytes (Lym). Automated localization and classification of these targets could support quantitative analysis of sputum cellular composition.

Conventional sputum cytology relies primarily on manual microscopy, which is time-consuming and affected by experience, fatigue, and sample quality. Automated image analysis has been applied to pathogen identification in sputum smears [5,6]. However, mucus, debris, uneven staining, dense overlap, and occlusion complicate these images, and small targets such as lymphocytes and bacteria can be confused with background debris, leaving both manual and conventional methods vulnerable to missed and false detections [7,8].

Deep-learning methods have been increasingly applied to medical imaging, demonstrating the feasibility of automatically localizing and classifying small targets [9-11]. Clinical sputum images differ from general object-detection datasets: cells frequently touch or occlude one another, scales vary markedly, and class frequencies are naturally imbalanced, which limits feature extraction and box localization. To address these limitations, the InternImage-RTMDet baseline incorporates an InternImage backbone. Edge-guided deformable adaptation strengthened boundary modeling, adaptive gated feature fusion accommodated sample-specific backgrounds, DyHead aligned features across scales, and the loss was modified to reduce long-tailed and missing-annotation effects. The resulting model, GRA-InternRTM, improves feature extraction and localization stability in complex backgrounds.

2. MATERIALS AND METHODS
2.1 Baseline Architecture and Suitability Analysis
InternImage feature extraction network. InternImage is a convolution-based vision foundation model that uses DCNv3 as its core operator [12]; its dynamic receptive fields combine local convolution with Transformer-like adaptive modeling. Its architecture is shown in Figure 1.

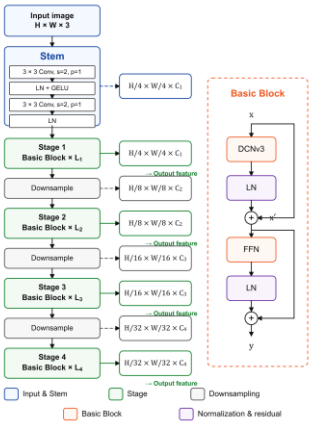


Figure 1. Hierarchical feature extraction network of InternImage (redrawn from Ref. [12]).

The stem layer encodes the input with two 3×3 convolutions (stride 2, padding 1) separated by layer normalization and GELU, reducing resolution to one-quarter while retaining low-level texture. Four stages then stack Basic Blocks with downsampling; as depth increases, resolution decreases and channels increase,

with earlier stages preserving cell-edge information and later stages modeling semantic context.

The Basic Block uses a two-branch Post-Norm structure, with separate residual paths for spatial and channel information. In the first submodule, DCNv3 processes the feature map. Learnable offsets Δp and sampling-point weights m dynamically adjust sampling locations according to local morphology. This operation improves feature capture near complex boundaries. Layer normalization then standardizes the features before a residual connection merges them with the original input. The computation is defined as follows:

$$\hat{\boldsymbol{x}} = \boldsymbol{x} + \mathrm{LN}\big(\mathrm{DCNv3}(\boldsymbol{x})\big) \quad \textbf{(1)}$$

The second submodule passes the output through a feed-forward network (FFN); layer normalization and a residual connection yield the final output:

$$y = \hat{x} + \mathrm{LN}\big(\mathrm{FFN}(\hat{x})\big) \quad \textbf{(2)}$$

In Eqs. (1) and (2), $\boldsymbol{x}$ denotes the input feature map, $\hat{\boldsymbol{x}}$ denotes the first submodule output and second submodule input, and y denotes the Basic Block output. DCNv3 provides Transformer-like global contextual modeling, while the FFN strengthens information exchange across channels. Their combination provides richer feature representations for downstream detection.

Baseline model for multiclass sputum cell detection. Given the sputum dataset's complex backgrounds, large scale variation, and severe overlap, a one-stage baseline, InternImage-RTMDet, was constructed using InternImage [12] as the backbone, PAFPN [13] as the neck, and RTMDet [14] as the detection head (Figure 2).



Figure 2. Baseline InternImage-RTMDet detection framework.

Sputum cells, especially epithelial cells, often show curled edges and irregular morphologies that fixed sampling handles poorly. DCNv3 [12] adjusts sampling through learnable offsets toward informative regions, PAFPN [13]adds a bottom-up path to preserve small-target information, and the decoupled RTMDet head separates classification from regression.

InternImage-RTMDet processes the input through the backbone; outputs from Stages 2–4 (224, 448, 896 channels) form a multiscale pyramid. PAFPN performs bidirectional aggregation and compresses each level to 256 channels. The decoupled head then predicts class scores and box regressions at each level for end-to-end detection.

Despite these advantages, limitations remain in clinical samples. Because DCNv3 offsets derive mainly from image features, sampling points may drift into backgrounds when boundaries have low contrast or overlapping cytoplasm. The balance between shallow texture and deep semantics varies among samples, yet conventional PAFPN adds multiscale features directly, which can dilute critical signals. The dataset is also long-tailed, with Lym and Bacteria representing only 3.63% and 6.93% of labels (Figure 3).

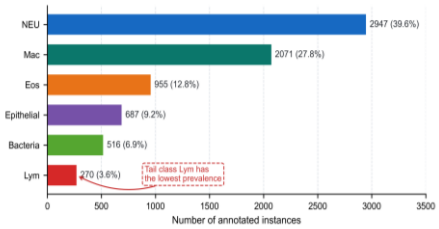


Figure 3. Distribution of cell-class annotations.

Incomplete annotations also generate erroneous negative gradients during training, limiting model performance.

To address these limitations, InternImage-RTMDet was modified through edge guidance, gated fusion, attention enhancement, and robust loss design. The resulting model is referred to as GRA-InternRTM.

2.2 GRA-InternRTM for Sputum Cell Detection

The overall architecture of GRA-InternRTM is shown in Figure 4.

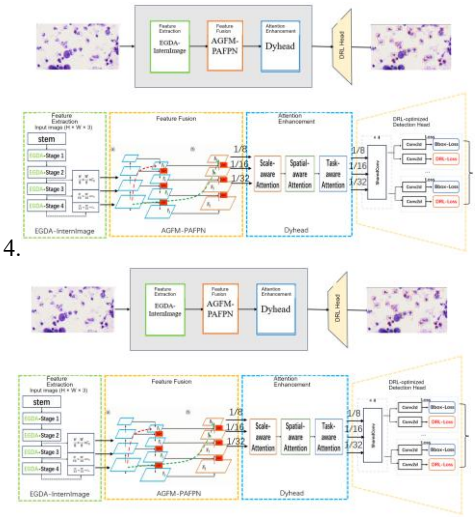


Figure 4. Overall architecture of GRA-InternRTM.

InternImage remains the backbone, with DCNv3 convolutions capturing cell morphology. EGDA is inserted into Stages 1–4 to reduce sampling drift: it extracts gradient priors with a Sobel operator and constrains offset generation, providing more stable boundary information.

During fusion, multiscale features enter PAFPN for bidirectional aggregation. Spatial gating weights dynamically adjust the fusion ratio according to each sample's dependence on shallow textures versus deep semantics, retaining local contours while incorporating high-level content.

An attention-enhancement module sequentially applies scale-, spatial-, and task-aware attention to realign feature maps across cell classes.

Finally, a DRL-optimized head reweights tail classes and downweights supervision from potentially unannotated targets.

Edge-Guided Deformable Adapter for boundary feature extraction. DCNv3 offsets are primarily content-driven and may drift into backgrounds under strong texture interference. Gradient magnitude stably represents object boundaries [15], and explicit edge priors improve contour sensitivity [16]. The Edge-Guided Deformable Adapter (EGDA) was therefore designed to constrain offset learning near cell boundaries.

EGDA introduces an interpretable low-level prior into deformable sampling: an explicit edge-extraction branch is added before DCNv3 offset computation (Figure 5).

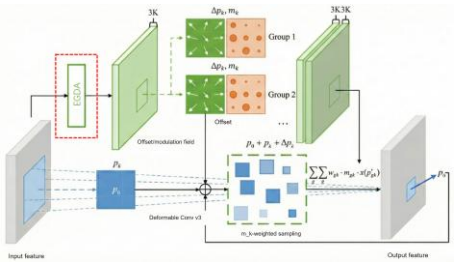


Figure 5. DCNv3 sampling constrained by EGDA (the DCNv3 sampling diagram was redrawn from Ref. [12]).

EGDA is the red dashed box in Figure 5. Before offset learning it extracts gradient information from the input features using fixed Sobel kernels:

$$K_x = [[-1,0,1],[-2,0,2],[-1,0,1]]$$
$$K_y = [[-1,-2,-1],[0,0,0],[1,2,1]]$$

These kernels are depthwise-convolved with the input features. This operation yields the horizontal and vertical gradient components G_x and G_y, respectively. Equation (3) then gives the gradient magnitude at position (x, y). Here, ϵ is a small perturbation term used to maintain numerical stability.

$$G = \sqrt{G_x^2 + G_y^2 + \epsilon} \quad (3)$$

The extracted gradient feature G is fused with the original input feature X. Their joint representation is supplied to the offset-generation layer. The offsets are generated as follows:

$$\Delta p = \sigma\left(\mathrm{Conv}_{off}([X;G])\right) \quad (4)$$

Thus, the offset Δp depends on both abstract semantic information and image gradients. Explicit gradient constraints on deformable-convolution sampling help reduce out-of-bounds sampling and drift into background regions in complex sputum images.

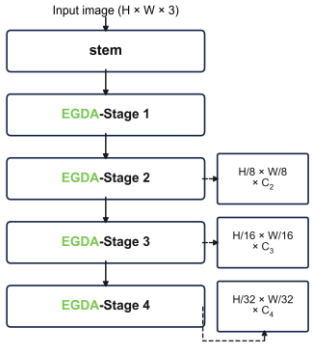


Figure 6. Feature extraction architecture incorporating EGDA (redrawn from Ref. [12]).

EGDA modifies DCNv3 in all four stages. After a three-channel image enters, Stage 1 first activates EGDA; gradient-corrected offsets then propagate through the later stages. Outputs from Stages 2–4 (224, 448, 896 channels) are passed to feature fusion.

Adaptive Gated Fusion Module for background adaptation. Conventional PAFPN adds hierarchical features element-wise, but the contribution of each level varies across samples; large semantic and resolution gaps limit direct fusion [17], and selective multi-kernel attention helps [18]. The Adaptive Gated Fusion Module (AGFM) was therefore introduced into PAFPN to replace fixed addition with gated weighting (Figure 7).

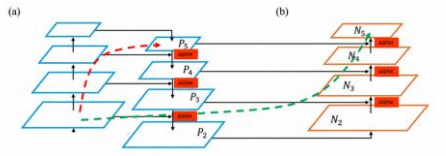


Figure 7. Architecture of the feature fusion module (the main PAFPN pathways were redrawn from Ref. [13]).

AGFM introduces a pixel-level gating mask M, computed as follows:

$$M = \sigma\left(\mathrm{Conv}_{1\times1}([F_{\mathrm{low}};F_{\mathrm{high}}])\right) \quad (5)$$

AGFM concatenates and maps shallow and deep features through a 1 × 1 convolution, normalizes the result with a sigmoid into weights in $[0,1]$, and fuses the features accordingly:

$$F_{\mathrm{out}} = M \odot F_{\mathrm{low}} + (1-M) \odot F_{\mathrm{high}} \quad (6)$$

AGFM receives the multiscale stream from Stages 2–4 and dynamically weights features at different resolutions along the PAFPN paths, retaining global semantics and local contours. It outputs three 256-channel maps at the extraction scales.

Dynamic Head for multiscale feature alignment and representation. Because cell sizes vary markedly across and within classes, the baseline exhibits uneven detection performance across target scales. The model incorporates the Dynamic Head (DyHead) of Dai et al. [19], which models level-, spatial-, and channel-wise attention independently (Figure 8).



Figure 8. Architecture of DyHead.

Scale-aware attention combines feature levels by target size; spatial-aware attention uses deformable sampling to focus on discriminative locations such as nuclear edges; and task-aware attention gates channels for classification versus localization. These operations are defined as follows:

$$\mathrm{Out} = \pi_C(\pi_S(\pi_L(F)\cdot F)\cdot F)\cdot F \quad (7)$$

Here, π_L denotes the attention-weighting operation along the feature-level dimension, π_S denotes spatial attention, and π_C denotes channel-wise task attention.

DyHead stacks the 1/8, 1/16, and 1/32 scale maps into a level–spatial–channel tensor. Scale-aware attention reallocates importance among levels, spatial-aware attention combines with EGDA's edge prior to sample discriminative regions, and channel-wise task attention separates classification and localization pathways.

Loss optimization for the detection head. Normal physiological variation produces unequal sputum cell proportions, and lymphocytes typically constitute a small proportion of cells in clinical samples [20]. Sampling variability and annotation difficulty further intensify this imbalance; Lym cells are small and resemble debris, making them particularly susceptible to missing annotations. Focal Loss addresses class imbalance by reweighting easy and difficult samples, thereby emphasizing rare and hard examples [21]. For label noise, Zhang et al. noted that mean absolute error (MAE) is noise-robust but may underperform on deep networks and large datasets. They proposed Generalized Cross Entropy (GCE) to combine properties of cross-entropy loss and MAE [22]. These ideas informed the design of Dual-Robust Learning loss (DRL), which comprises two complementary branches; the first introduces the class-specific weight α_c into Focal Loss to mitigate the long-tailed distribution:

$$L_{focal} = -\alpha_c(1-p_t)^{\gamma}\log(p_t) \quad (8)$$

Here, p_t is the predicted probability of the true class, and α_c is the weight assigned to cell class c. Weights were set to 1 for the head classes NEU and Mac, 4 for the tail class Lym, and 2 for the remaining classes. This weighting increases the contribution of rare classes during backpropagation. The second branch addresses incomplete annotations through a dynamic factor based on prediction confidence. The final DRL loss is defined as follows:

$$L_{DRL} = \sum_i m_i \cdot L_{focal,i} \quad \textbf{(9)}$$

Here, m_i is a confidence-based downweighting factor, calculated as follows:

$$m_i = \begin{cases} \exp\left(-\frac{(p_i - \tau)^2}{\beta}\right) & \text{if } y_i = \text{Background and } p_i > \tau \\ 1 & \text{otherwise} \end{cases} \quad \textbf{(10)}$$

Here, y_i is the ground-truth label for the current region, p_i is the predicted target confidence, and τ is a preset threshold. When an unlabeled region is predicted as a cell with high confidence, m_i approaches zero. The model therefore avoids penalizing that sample as a strong negative. As confidence decreases, m_i increases, strengthening the sample's contribution to training. This weighting is intended to reduce erroneous negative gradients from potentially missing annotations.

The resulting DRL-optimized detection head retains RTMDet's decoupled design while adapting parameter sharing to the new loss (Figure 9).

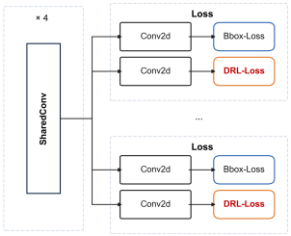


Figure 9. Detection head incorporating DRL (redrawn from Ref. [14]).

The DRL head stacks four shared convolutional layers with parallel branches for the three DyHead scales; DRL replaces the original classification loss in each branch to improve rare-class learning and reduce missing-annotation influence.

These components act at different points in the detection pipeline rather than repeatedly modifying the same representation. EGDA supplies gradient information to offset generation during feature extraction, whereas AGFM adjusts the relative contributions of hierarchical features during fusion. DyHead subsequently refines the fused representation across scale, spatial, and task dimensions. DRL operates through the training objective, changing the influence of classes and candidate regions on parameter updates rather than introducing another feature-fusion operation. This separation provides the rationale for combining the components: changes to feature sampling, feature aggregation, and supervision address different aspects of the detection task. Whether their combination improves performance must nevertheless be assessed empirically through the ablations, rather than inferred from their intended functions alone.

2.3 Experimental Setup

Dataset. The data were derived from clinical sputum specimens collected by the respiratory department of a tertiary hospital. Samples were stained with the Wright-Giemsa method; the strong contrast between nuclei and cytoplasm facilitates feature extraction. A high-throughput scanner generated whole-slide images (.svs). In total, 16 specimens from different individuals were included. Cropping the whole-slide images yielded 237 images containing 7,446 annotated instances across six classes. The cropped images were divided into training and test sets at an 8:2 ratio. All cropped images originating from the same specimen were assigned exclusively to one set, ensuring no specimen overlap.

Training used images at their original resolution of 2,112 × 1,184 pixels with weak augmentation (random resizing, cropping, and flipping) and ClassBalancedDataset sampling. Models were trained for 80 epochs with a batch size of 4 using AdamW (initial learning rate 1.5×10^{-4}, weight decay 0.05) with linear warmup for 1,000 iterations; MultiStepLR reduced the learning rate by 0.1 at epochs 70 and 75.

Table 1. Dataset summary: (A) basic characteristics and (B) class distributions in the training and test sets.

A | Basic dataset characteristics

Parameter	Description/value
Number of sputum specimens	16
Image format	PNG/JPG (three-channel RGB)
Number of images	237
Annotated instances	7,446
Mean targets per image	31.42
Annotation tool	LabelMe
Annotation format	MS COCO
Number of classes	6

B | Class distributions

Class	Training set	Test set	Total	Training:test ratio
NEU	2274	673	2947	3.4:1
Eos	735	220	955	3.3:1
Mac	1701	370	2071	3.4:1
Lym	210	60	270	3.5:1
Epithelial	544	143	687	3.8:1
Bacteria	428	88	516	4.9:1

Experimental environment and parameter settings. Model training and validation were conducted on Linux using a single GPU and the MMDetection framework. The hardware and software configurations are listed in Table 2.

Table 2. Hardware and software environment.

Environment	Configuration
Operating system	Ubuntu 22.04
CPU	24 vCPU AMD EPYC 9754 (128-Core Processor)
GPU	NVIDIA vGPU-32GB
Programming language	Python 3.9.23
Deep-learning framework	PyTorch 2.7.0
CUDA version	12.6
MMCV version	2.1.0
MMDetection version	3.3.0

Evaluation metrics. Performance was evaluated from three perspectives: detection quality (COCO-style mAP), detection capability (mAP50), and classification reliability (Recall and matched classification accuracy, AccMatched).

For six-class detection, a true positive (TP) is a predicted box matched to a ground-truth box at the specified IoU threshold; a false positive (FP) is an unmatched prediction, including background, duplicate, and misclassified detections; and a false negative (FN) is a ground-truth target undetected or unmatched below the threshold. Mean average precision (mAP) averages AP across IoU thresholds from 0.50 to 0.95, reflecting localization quality. It is calculated as follows:

$$mAP = \frac{1}{|\mathcal{C}|}\sum_{c\in\mathcal{C}}\left(\frac{1}{|\mathcal{T}|}\sum_{\tau\in\mathcal{T}} AP_c^{\tau}\right) \quad (11)$$

Here, $\mathcal{C}$ is the number of classes, and $\mathcal{C}$ = 6 for this dataset. The set of IoU thresholds $\mathcal{T}$ contains 0.50, 0.55, 0.60, …, 0.95 under the COCO protocol. The term AP_c^{τ} is the average precision for a specific class and IoU threshold, equal to the area under the precision-recall curve. COCO uses 101-point interpolation, calculated as follows:

$$AP_c^{\tau} = \frac{1}{101}\sum_{r\in(0,0.01,\ldots,1.00)}\left(\max_{\hat{r}\geq r} p(\hat{r})\right) \quad (12)$$

Here, r denotes recall sampling points from 0 to 1 at increments of 0.01, and p is precision at the corresponding recall. The mAP50 metric is mean average precision at an IoU threshold of 0.50. It is less stringent than mAP and reflects detection capability at a more permissive overlap criterion, which is useful when sputum cell boundaries overlap or appear blurred. It is calculated as follows:

$$mAP_{50} = \frac{1}{|\mathcal{C}|}\sum_{c\in\mathcal{C}} AP_c^{0.50} \quad (13)$$

The mAP and mAP50 metrics primarily reflect bounding-box quality and localization accuracy. Recall and AccMatched were additionally evaluated at IoU = 0.5. Recall is the proportion of annotated cells successfully detected and is clinically relevant because missed detections may leave important cells undetected. It is calculated as follows:

$$Recall = \frac{TP}{TP + FN} \quad (14)$$

$Acc_Matched$ was used to quantify correct class predictions among successfully matched samples, as follows:

$$Acc_Matched = \frac{TP}{TP + FP(match)} \quad (15)$$

False positives comprise background detections, classification errors, and duplicate predictions. In Eq. (15), $FP(match)$ represents FP after background detections have been excluded. Unlike Precision, AccMatched measures the proportion of correct classifications among predictions matched to annotated targets. It therefore characterizes class confusion rather than background rejection.

3. RESULTS

3.1 Ablation Studies

Overall detection performance. Table 3 summarizes the module-combination ablations. EGDA was assessed both by addition to the InternImage-RTMDet baseline (B) and by removal from the full GRA-InternRTM configuration (F), whereas AGFM, DyHead, and DRL were evaluated individually and in selected combinations.

The ablations (Table 3) show that EGDA produced consistent gains: adding it increased all four metrics, while removing it from the full model reduced mAP, mAP50, Recall, and AccMatched by 0.011, 0.010, 0.0257, and 0.0137. AGFM and DRL each increased mAP50, Recall, and AccMatched but slightly reduced mAP alone, while DyHead raised mAP and Recall but reduced AccMatched; it also raised mAP_m from 0.274 to 0.297 and mAP_l from 0.256 to 0.288, with the full model reaching 0.337 and 0.347. The joint use of all components achieved the best aggregate performance, suggesting complementary effects.

The combined configuration also outperformed the individual additions. AGFM, DyHead, and DRL together yielded an mAP of 0.354, compared with 0.291, 0.315, and 0.297 when each was added separately to the baseline. Adding EGDA to this combination increased mAP to 0.365. The individual configurations should also be interpreted across metrics: the mAP decreases accompanying AGFM or DRL were not accompanied by decreases in all other measures. An improvement in matching or classification performance therefore did not necessarily imply an improvement in detection precision averaged over multiple localization thresholds. The full configuration provided a better aggregate balance in the reported comparison, supporting assessment of the components within the combined framework rather than selection using one isolated metric. However, the selected configurations do not establish all pairwise interactions or demonstrate that their effects are additive.

Table 3. Module-combination ablation results for GRA-InternRTM.

EGDA	AGFM	DyHead	DRL	mAP	mAP50	Recall	AccMatched
-	-	-	-	0.302	0.562	0.8026	0.8804
√	-	-	-	**0.334**	0.571	0.8138	0.9014
-	√	-	-	0.291	0.569	0.8247	**0.9047**
-	-	√	-	0.315	0.573	0.8153	0.8601
-	-	-	√	0.297	**0.578**	**0.8253**	0.8927
-	√	√	-	0.326	0.587	0.8449	0.9405
-	-	√	√	0.342	0.577	0.8438	0.9493
-	√	-	√	0.313	0.591	0.8341	0.9124
-	√	√	√	**0.354**	**0.612**	0.8505	0.9480
√	√	-	√	0.345	0.595	**0.8651**	**0.9503**
√	√	√	√	**0.365**	**0.622**	**0.8762**	**0.9617**

Note: '√' indicates inclusion; '–' indicates exclusion.

Class-specific performance. Class-specific effects under the long-tailed distribution were further examined. Results for the baseline, baseline plus DRL, and full model are reported in Table 4.

Table 4. Class-specific detection performance for the six cell classes.

Method	Category	mAP	mAP50	Recall	AccMatched
Baseline	NEU	0.375	0.726	**0.9554**	0.8748
	Mac	**0.480**	**0.765**	0.8621	0.9593
	Eos	0.334	0.621	0.6809	**0.9792**
	Epithelial	0.237	0.462	0.8252	0.7813
	Bacteria	0.261	0.546	0.6295	0.9605
	Lym	0.125	0.255	0.5080	0.7333
	Overall	0.302	0.562	0.8026	0.8804
Baseline + DRL	NEU	0.331	0.641	**0.8732**	0.8817
	Mac	**0.435**	**0.774**	0.8347	0.9471
	Eos	0.325	0.625	0.7147	0.9679
	Epithelial	0.286	0.617	0.8334	0.8247
	Bacteria	0.274	0.531	0.7173	**0.9817**
	Lym	0.131	0.283	0.5740	0.8133
	Overall	0.297	0.578	0.8253	0.8927
Full	NEU	0.392	0.719	**0.9159**	0.9505
	Mac	**0.559**	**0.825**	0.8757	0.9446
	Eos	0.390	0.633	0.7364	0.9586
	Epithelial	0.422	0.677	0.8462	0.9603
	Bacteria	0.296	0.568	0.6877	**1.0000**
	Lym	0.136	0.311	0.5333	0.8696
	Overall	0.365	0.622	0.8762	0.9617

DRL redistributed performance rather than improving every metric. The clearest gains occurred for underrepresented classes: Lym Recall increased from 0.5080 to 0.5740 and AccMatched from 0.7333 to 0.8133, while Epithelial mAP increased from 0.237 to 0.286; aggregate mAP decreased slightly from 0.302 to 0.297.

The gains for Lym coincided with reduced performance for the abundant NEU class: adding DRL reduced NEU mAP from 0.375 to 0.331 and Recall from 0.9554 to 0.8732. This pattern is consistent with a redistribution of performance across classes, rather than a uniform improvement. The results alone do not isolate the contributions of class weighting and confidence-based downweighting within DRL.

The full model improved most class-metric combinations, with the largest mAP gains for Epithelial (0.237 to 0.422) and Mac (0.480 to 0.559). Lym AccMatched improved to 0.8696, but its final mAP remained 0.136, showing the rarest class remained difficult.

Lym remained a limitation despite its higher matched classification accuracy. Its Recall with the full model was 0.5333, below the 0.5740 obtained with baseline plus DRL, although AccMatched increased from 0.8133 to 0.8696. More accurate classification among matched predictions therefore did not imply that more annotated Lym targets were detected. Only 60 Lym instances were included in the test set, which further limits conclusions about performance on this rare class.

3.2 Comparative Experiments

GRA-InternRTM was compared with two-stage detectors (Fast R-CNN [23], Cascade R-CNN [24], Mask R-CNN [25]), one-stage detectors (YOLOv8 [26], YOLOv10 [27]), and a Swin-Transformer-based detector [28], all under the same dataset split and conditions (Table 5).

GRA-InternRTM attained the highest mAP, mAP50, and AccMatched among compared models. Its Recall (0.8762) was 0.0016 below that of Fast R-CNN, while its mAP, mAP50, and AccMatched exceeded those of Fast R-CNN by 0.014, 0.024, and 0.0308.

The comparison indicates a trade-off rather than superiority on every metric. The 0.0016 Recall difference favored Fast R-CNN, whereas the other three reported metrics favored GRA-InternRTM. The distinction between mAP and mAP50 provides additional context. Whereas mAP50 uses one relatively permissive overlap threshold, mAP summarizes detection precision across multiple thresholds, including stricter localization requirements. Leading both measures therefore indicates an advantage under the permissive criterion and in the aggregate evaluation across thresholds; it does not establish superiority at every individual threshold. The slightly lower Recall also shows that the best aggregate precision did not correspond to the highest proportion of annotated targets detected. AccMatched evaluates classification among matched predictions and does not measure rejection of background detections; its improvement should therefore not be interpreted as evidence of fewer background false positives. Together, these metrics describe distinct aspects of performance and should not be treated as interchangeable. These numerical differences have not been tested for statistical significance.

Table 5. Performance comparison of detection models.

Model	mAP / mAP_{50}	Recall	$Acc_{Matched}$
Fast R-CNN	0.351 / 0.598	0.8778	0.9309
Cascade R-CNN	0.246 / 0.499	0.8063	0.8541
Mask R-CNN	0.337 / 0.597	0.8295	0.9402
YOLOv8	0.292 / 0.579	0.8095	0.8523
YOLOv10	0.317 / 0.614	0.7841	0.8247
Swin Transformer	0.332 / 0.604	0.8398	0.9001
GRA-InternRTM	**0.365 / 0.622**	**0.8762**	**0.9617**

4. DISCUSSION

The ablations indicate that GRA-InternRTM's performance arose from the combined design. EGDA improved all aggregate metrics, whereas AGFM and DRL did not independently improve every metric. When added alone, DyHead reduced matched classification accuracy. Because a single split without repeated runs was used, statistical significance remains to be established.

A possible explanation for the combined gains is that edge guidance, adaptive feature fusion, and attention address different aspects of the representation. Edge constraints may help sampling near cell boundaries, while fusion and attention may help reconcile features across resolutions. This interpretation is consistent with the model design and observed ablations, but direct evidence from sampling-offset visualization or feature analysis is not available. Similarly, the DRL results do not independently establish robustness to missing annotations, because the class-weighting and confidence-weighting branches were not separately evaluated in the reported experiments.

For the sputum-cell detection task, locating a target and assigning its class are related but distinct requirements. A cell that is not detected cannot contribute to classification performance among matched predictions, so a high AccMatched value cannot compensate for missed targets. This distinction is particularly relevant to the Lym results, where better matched classification did not coincide with the highest Recall. Aggregate metrics are useful for comparing complete detection frameworks, while class-specific results reveal whether improvements extend to the less represented categories. Interpreted together, the findings support the combined use of boundary information, multiscale representations, and weighted supervision on the evaluated dataset, rather than a claim that every cell class or metric benefited uniformly.

This study was limited to 16 specimens and 237 cropped images from a single center, an internal test set, and potentially incomplete annotations. Multicenter external validation and repeated experiments are needed before broader use.

5. CONCLUSION

Automated sputum cell detection may provide a basis for more consistent quantitative analysis of microscopy images. This study presents GRA-InternRTM for blurred boundaries, heterogeneous backgrounds, scale variation, and class imbalance in cropped sputum images. The model combines EGDA for edge-informed feature extraction and AGFM for adaptive multiscale fusion. DyHead provides scale-, spatial-, and task-aware representation, while DRL reweights underrepresented classes and downweights potential annotation noise.

On the internal test set, the full model yielded 0.365 mAP, 0.622 mAP50, 0.8762 Recall, and 0.9617 AccMatched. Its mAP, mAP50, and AccMatched exceeded those of all comparison models, while Recall remained close to the highest comparison value. These findings support technical feasibility on the current dataset but do not demonstrate generalizability or clinical utility.

Future work should broaden the sources and distribution of samples, conduct external validation across additional centers and slide-preparation conditions, and further evaluate inference efficiency, robustness, and the feasibility of lightweight deployment.

6. ACKNOWLEDGMENTS

This research was supported by the Natural Science Foundation of Jiangxi Province (Grant No. 20232BAB212008).

7. AUTHOR CONTRIBUTIONS

Y. Liu: validation, writing: original draft; W. Yi: methodology, validation, formal analysis; Y. Feng: supervision, conceptualization, writing: review and editing; M. Zhang and D. Gai: investigation, data curation; W. Zhang: writing: review and editing; X. Zhong: Writing ,editing.

8. CONFLICTS OF INTEREST

The authors declare no conflicts of interest.

9. DATA AVAILABILITY STATEMENT

The raw data are not publicly shared due to patient confidentiality and research ethics; they are available from the corresponding author upon reasonable request.

10. REFERENCES

[1] State Council of the People's Republic of China, “Opinions of the State Council on Deepening the Implementation of the ‘Artificial Intelligence Plus’ Initiative,” Gazette of the State Council of the People's Republic of China, no. 25 (2025): 15-19, [in Chinese],

[2] N. Panakkal, A. Lekshmi, V. V. Saraswathy, et al., “Effective Lung Cancer Control: An Unaccomplished Challenge in Cancer Research,” CytoJournal 20 (2023): 16, doi: 10.25259/Cytojournal_36_2022.

[3] R. H. Pan, J. Yin, Y. Y. Wu, et al., “Association of Sputum Viscosity With Disease Severity and Prognosis in Adults With Mycoplasma pneumoniae Pneumonia,” Infectious Disease Information 39, no. 1 (2026): 80-85, [in Chinese], doi: 10.3969/j.issn.1007-8134.2026.01.013.

[4] L. Qin, P. G. Gibson, J. L. Simpson, et al., “Dysregulation of Sputum Columnar Epithelial Cells and Products in Distinct Asthma Phenotypes,” Clinical & Experimental Allergy 49, no. 11 (2019): 1418-1428, doi: 10.1111/cea.13452.

[5] K. S. Mithra and W. R. Sam Emmanuel, “Automated Identification of Mycobacterium Bacillus From Sputum Images for Tuberculosis Diagnosis,” Signal, Image and Video Processing 13, no. 8 (2019): 1585-1592, doi: 10.1007/s11760-019-01509-1.

[6] K. S. Mithra, “Enhanced Fuzzy Gaussian Networks for Sputum Image Based Mycobacterium Detection,” The Imaging Science Journal 71, no. 4 (2023): 313-322, doi: 10.1080/13682199.2023.2167553.

[7] R. Khutlang, S. Krishnan, A. Whitelaw, et al., “Automated Detection of Tuberculosis in Ziehl-Neelsen-Stained Sputum Smears Using Two One-Class Classifiers,” Journal of Microscopy 237, no. 1 (2010): 96-102, doi: 10.1111/j.1365-2818.2009.03308.x.

[8] Z. Jan, M. Rafiq, H. Muhammad, et al., “Detection of Tuberculosis Bacteria in Sputum Slide Image Using Morphological Features,” in Beyond Databases, Architectures and Structures (Cham, Switzerland: Springer, 2015), 408-414, doi: 10.1007/978-3-319-18422-7_36.

[9] Z. Le, Y. Liang, X. Hu, T. Qiu, and P. Xu, “A Risk Stratification Study of Ultrasound Images of Thyroid Nodules Based on Improved DETR,” International Journal of Imaging Systems and Technology 34, no. 6 (2024): e23219, doi: 10.1002/ima.23219.

[10] M. Salvi, N. Michielli, K. M. Meiburger, et al., “cyto-Knet: An Instance Segmentation Approach for Multiple Myeloma Plasma Cells Using Conditional Kernels,” International Journal of Imaging Systems and Technology 34, no. 1 (2024): e22984, doi: 10.1002/ima.22984.

[11] V. Biradar and K. C. Gull, “Multiple Object Detection and Tracking Using Deep Learning Framework with Non-Maximum Suppression,” International Journal of Intelligent Engineering and Systems 18, no. 1 (2025): 106-117, doi: 10.22266/ijies2025.0229.09.

[12] W. Wang, J. Dai, Z. Chen, et al., “InternImage: Exploring Large-Scale Vision Foundation Models With Deformable

Convolutions," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada (Piscataway, NJ: IEEE, 2023), 14408-14419, doi: 10.1109/CVPR52729.2023.01385.

[13] S. Liu, L. Qi, H. Qin, et al., "Path Aggregation Network for Instance Segmentation," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA (Piscataway, NJ: IEEE, 2018), 8759-8768, doi: 10.1109/CVPR.2018.00913.

[14] C. Lyu, W. Zhang, H. Huang, et al., "RTMDet: An Empirical Study of Designing Real-Time Object Detectors," arXiv preprint arXiv:2212.07784 (2022), doi: 10.48550/arXiv.2212.07784.

[15] R. C. Gonzalez and R. E. Woods, Digital Image Processing, 4th ed. (New York, NY: Pearson, 2018), ISBN 978-0-13-335672-4.

[16] T. Takikawa, D. Acuna, V. Jampani, et al., "Gated-SCNN: Gated Shape CNNs for Semantic Segmentation," in Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, South Korea (Piscataway, NJ: IEEE, 2019), 5228-5237, doi: 10.1109/ICCV.2019.00533.

[17] Z. Zhang, X. Zhang, C. Peng, et al., "ExFuse: Enhancing Feature Fusion for Semantic Segmentation," in Computer Vision - ECCV 2018, Munich, Germany (Cham, Switzerland: Springer, 2018), 273-288, doi: 10.1007/978-3-030-01249-6_17.

[18] X. Li, W. Wang, X. Hu, et al., "Selective Kernel Networks," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA (Piscataway, NJ: IEEE, 2019), 510-519, doi: 10.1109/CVPR.2019.00060.

[19] X. Dai, Y. Chen, B. Xiao, et al., "Dynamic Head: Unifying Object Detection Heads With Attentions," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA (Piscataway, NJ: IEEE, 2021), 7369-7378, doi: 10.1109/CVPR46437.2021.00729.

[20] J. Belda, R. Leigh, K. Parameswaran, et al., "Induced Sputum Cell Counts in Healthy Adults," American Journal of Respiratory and Critical Care Medicine 161, no. 2 (2000): 475-478, doi: 10.1164/ajrccm.161.2.9903097.

[21] T.-Y. Lin, P. Goyal, R. Girshick, et al., "Focal Loss for Dense Object Detection," IEEE Transactions on Pattern Analysis and Machine Intelligence 42, no. 2 (2020): 318-327, doi: 10.1109/TPAMI.2018.2858826.

[22] Z. Zhang and M. R. Sabuncu, "Generalized Cross Entropy Loss for Training Deep Neural Networks With Noisy Labels," in Advances in Neural Information Processing Systems 31, Montreal, Canada (Red Hook, NY: Curran Associates, 2018), 8778-8788,

[23] R. Girshick, "Fast R-CNN," in Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile (Piscataway, NJ: IEEE, 2015), 1440-1448, doi: 10.1109/ICCV.2015.169.

[24] Z. Cai and N. Vasconcelos, "Cascade R-CNN: Delving Into High Quality Object Detection," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA (Piscataway, NJ: IEEE, 2018), 6154-6162, doi: 10.1109/CVPR.2018.00644.

[25] K. He, G. Gkioxari, P. Dollar, et al., "Mask R-CNN," IEEE Transactions on Pattern Analysis and Machine Intelligence 42, no. 2 (2020): 386-397, doi: 10.1109/TPAMI.2018.2844175.

[26] G. Jocher, A. Chaurasia, and J. Qiu, "Ultralytics YOLOv8," version 8.0.0 (2023),

[27] A. Wang, H. Chen, L. Liu, et al., "YOLOv10: Real-Time End-to-End Object Detection," Advances in Neural Information Processing Systems 37 (2024): 107984-108011, doi: 10.52202/079017-3429.

[28] Z. Liu, Y. Lin, Y. Cao, et al., "Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows," in Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada (Piscataway, NJ: IEEE, 2021), 9992-10002, doi: 10.1109/ICCV48922.2021.00986.