

Mitigating the Effect of Resizing in YOLO for Small Object Detection Applied to Raptor Detection

Enol GARCÍA GONZÁLEZ^{1*}, Laura MENÉNDEZ GARCÍA¹, José R. VILLAR¹, Javier SEDANO²

¹ Computer Science Department, University of Oviedo, C. Jesús Arias de Velasco, Oviedo, 33005, Spain
garciaenol@uniovi.es (*Corresponding author), uo283055@uniovi.es, villarjose@uniovi.es

² Institute of Technology of Castilla y Leon, C. López Bravo, 70, Burgos, 09001, Spain
javier.sedano@itcl.es

Abstract: Single-stage object detectors such as YOLO are widely used due to their efficiency and simplicity, but they feature well-known limitations when dealing with small object detection in high-resolution images. A key factor contributing to this issue is the global resizing of input images, which can severely reduce the effective spatial resolution of small targets. This problem is especially relevant in outdoor monitoring scenarios, such as the detection of birds of prey in free-range poultry farms, where the objects of interest often appear distant and occupy only a small fraction of the analysed scene. This paper proposes a region-based preprocessing strategy which was designed in order to mitigate the adverse effects of image resizing in the context of YOLO-based detectors. This approach consists in partitioning the original image into multiple localized regions that are processed independently by the detection model, allowing small objects to be analysed at a higher relative resolution without altering the network architecture or the training procedure. This method is intended to be model-agnostic and easily integrable into the existing YOLO-based pipelines. The proposed strategy is applied for detecting birds of prey in outdoor farm environments, providing a structured framework for improving small object representation in single-stage detectors. This work aims to contribute a practical preprocessing solution that addresses a fundamental limitation of YOLO in scenarios dominated by small and sparsely distributed objects.

Keywords: Object detection, Bird detection, YOLO.

1. Introduction

Automatic bird detection in open environments has become a topic of increasing interest in monitoring and security applications, particularly in scenarios where the presence of birds of prey may pose an economic or environmental risk. In the context of free-range poultry farms, the early detection of this type of bird is essential to minimize livestock attacks and to improve the prevention systems based on visual surveillance.

In recent years, deep learning-based object detection models have demonstrated a remarkable performance across a wide range of domains. In particular, single-stage detectors such as YOLO have gained widespread adoption due to their high computational efficiency and their ability to perform near real-time detection. These characteristics make them attractive candidates for continuous monitoring systems deployed in real-world environments.

Nevertheless, despite their overall success, YOLO-based models exhibit well-known limitations when confronted with scenarios dominated by small objects within high-resolution images. This issue is especially relevant in outdoor surveillance applications, where objects of interest often appear at large distances from the camera and occupy only a very small fraction of the overall image area. In such cases, the global resizing of input images

required by the model architecture can lead to a significant loss of spatial information, negatively affecting the object detection performance.

This work continues a line of research initiated in a previous contribution, in which the detection of birds of prey was addressed through a comparative analysis of single-stage and two-stage detectors using the AirBirds dataset. In that study, it was observed that while two-stage detectors achieved a superior performance in terms of precision, YOLO-based models exhibited a significantly lower performance in the detection of small objects. These results are consistent with the findings reported in the literature for other domains where YOLO has been applied to small object detection tasks, including road damage inspection (Jeong & Kim, 2022), electronic component quality control (Klco et al., 2023), and tree detection in aerial imagery (Moraes et al., 2025), where similar limitations have been documented. Taken together, these results highlight a structural limitation of single-stage detectors that extends beyond a single application domain.

Based on these observations, this paper proposes an extension of the previous study of García, Villar & Sedano (2026) focused on analyzing and mitigating one of the main causes of this reduced

performance: the effect of global input image resizing on the representation of small objects. Rather than modifying the internal architecture of the detector or introducing complex changes to the training procedure, a preprocessing strategy based on the spatial partitioning of the original image into smaller regions is proposed.

The central hypothesis of this work is that, by processing localized image regions independently, small objects can be analyzed at a higher effective resolution, improving their representation in the feature maps and, consequently, the overall detection performance. This strategy preserves the efficiency advantages of YOLO-based models while allowing their seamless integration into the existing detection pipelines.

The remainder of this paper is organized as follows. Section 2 describes the analyzed problem and its scientific relevance. Section 3 presents the proposed solution and details its implementation. Subsequently, Section 4 describes the experimental setup used for validating the proposed method, along with the materials required to conduct the experiments. Section 5 presents the experimental results together with an analytical discussion. Finally, Section 6 concludes this work and outlines the directions for future research.

2. Problem Analysis: Small Objects in AirBirds

This work continues the research presented in a previous contribution of García, Villar & Sedano (2026). The main objective of that study was to evaluate the behavior of both approaches (single-stage and two-stage) in a scenario characterized by the presence of small and frequently distant objects within high-resolution images. For this purpose, the AirBirds dataset was used, whose main challenge lies in the reduced size of the annotations. In this context, the experimental results revealed a significant performance difference between the two model families.

In particular, two-stage detectors, and especially Faster R-CNN in its more complex variants, achieved notably higher average precision values than those obtained by the evaluated versions of YOLO. In contrast, single-stage models exhibited a considerably lower performance, revealing

difficulties in correctly detecting birds when they occupied a very small fraction of the image.

This behavior highlighted a relevant limitation of single-stage detectors in scenarios dominated by small objects (Dicu et al., 2024; Moraes et al., 2025). While two-stage models generate region proposals that are subsequently refined and classified, models such as YOLO perform object detection and classification simultaneously over feature maps with a reduced spatial resolution, which can negatively affect the representation of small objects.

The problem of small object detection using YOLO is widely recognized in the literature (Jeong & Kim, 2022; Klco et al., 2023; Shewajo & Fante, 2023; Benjumea et al., 2021), as many authors have reported that YOLO can be particularly problematic when dealing with small elements in large images.

From an architectural perspective, this limitation can be explained by the design of the model itself. YOLO (Wang, Yeh & Liao, 2024; Jocher & Qiu, 2024; Jocher, Chaurasia & Qiu, 2023) is considered a single-stage detector.

Single-stage detectors such as YOLO perform bounding box prediction and classification simultaneously, directly over feature maps extracted from the full image. That is, the model receives the image resized to a fixed input size (e.g. 640×640 pixels) and after a series of convolutional operations and spatial reduction stages (downsampling), it produces dense predictions over a grid of cells. Each cell is responsible for predicting objects whose spatial representation is associated with that region of the feature map.

This design offers significant advantages in terms of computational efficiency and inference speed. However, it requires that all relevant spatial information be preserved throughout the initial resizing process and the successive internal resolution reductions of the model. When the original image is of high resolution and the objects occupy a very small proportion of the total area, global resizing can further reduce their effective size in pixels. As a result, the object may be represented by an extremely low number of activations in the

intermediate feature maps, hindering both the accurate localization and classification.

In relative terms, if a bird initially occupies a very small percentage of the total image area, this proportion is preserved after resizing to the model's input size, but the absolute number of pixels representing it decreases. Subsequently, the downsampling operations inherent to the architecture further reduce the spatial resolution of the feature maps. In this scenario, the object may be represented by a very weak signal, close to the background noise, which negatively affects the discriminative capability of the detector.

In contrast, two-stage detectors (Yu & Leung, 2025), such as Faster R-CNN (Ren et al., 2015), follow a different strategy. In a first stage, they generate region proposals that identify potential object locations. These regions are then processed more locally in a second stage, where the bounding boxes are refined and the final classification is performed. Although the image also passes through an initial convolutional network, the proposal mechanism allows the processing to focus on subregions where the object occupies a larger relative proportion of the analyzed area. In other words, the object no longer competes with the entire image background and is instead evaluated within a spatial window more closely adjusted to its size.

This difference is particularly relevant in scenarios dominated by small and sparsely distributed objects. While in the case of a single-stage detector the model must directly identify very small objects within a large global context, in the case of a two-stage detector the region proposal process acts as a spatial focusing mechanism that facilitates a subsequent discrimination.

In the light of this comparison (Aboyomi & Daniel, 2023), it is reasonable to hypothesize that the low performance observed for YOLO in the preliminary study may be closely related to the combined effect of global input resizing and the progressive internal resolution reduction of the model. In particular, when objects represent an extremely small fraction of the original image, the resizing process can constitute a structural bottleneck that limits the information available for their detection.

3. Proposed Method: Image Tiling for YOLO Enhancement

In this work, a preprocessing strategy is proposed to mitigate the negative impact of global image resizing in single-stage detectors such as YOLO. The core idea is to modify the way visual information is presented to the employed model, without altering its internal architecture (Lu et al., 2020) or its training procedure, with the aim of improving the effective representation of small objects. While the use of spatial partitioning has been previously explored, Jeong & Kim (2022) applied tiling at inference time for road damage detection, and Shewajo & Fante (2023) employed a similar scheme for microscopic malaria image analysis, these approaches are domain-specific and limited to inference. The method proposed in this paper differs from other method in that tiling is applied during both training and inference, a configurable overlap is incorporated to prevent object fragmentation at tile boundaries, and the approach is designed to be model-agnostic across different YOLO architectures.

The proposed strategy is based on partitioning the original image into multiple smaller spatial regions (Jeong & Kim, 2022; Shewajo & Fante, 2023), which are processed independently by the detector. In this way, each subimage contains a more localized portion of the scene, where the objects of interest occupy a larger relative proportion of the total area. By applying the resizing required by the model to these regions instead of the full image, a greater amount of spatial details is preserved in small objects, increasing their effective representation in the feature maps.

The proposal is grounded in the idea of increasing the effective resolution of small objects without the need to modify the detector architecture or its training procedure. Rather than processing the entire image resized to a fixed input size, a prior spatial decomposition is introduced that allows visual information to be presented to the model in a more favorable manner for small object detection.

Partitioning the original image into smaller regions has a direct effect on the relationship between object size and input image size. By operating

on subimages, objects of interest occupy a larger relative proportion of the analyzed area, which enhances their representation in the feature maps generated by the network. Although the relative scale within each subimage is preserved, this spatial redistribution reduces the negative impact of global resizing and the successive resolution reduction operations inherent to the model.

The use of overlap between adjacent regions plays a key role as regards the robustness of the method. This overlap prevents objects located near region boundaries from being fragmented or lost during the partitioning process, ensuring that the relevant instances appear complete in at least one subimage. Additionally, it introduces controlled spatial redundancy, increasing the probability of detection in critical areas without increasing the complexity of the detector.

Another relevant aspect is the prior adjustment of the original image size to ensure its compatibility with the partitioning scheme. This step allows the generation of subimages with homogeneous dimensions, avoiding incomplete or irregular regions that could introduce noise during training or inconsistencies during inference. The homogeneity of the resulting regions facilitates the integration of the method into the standard YOLO-based pipelines.

Algorithm 1 describes the specific image tiling proposal presented in this work. The procedure begins by adjusting, if necessary, the size of the original image to ensure that partitioning can be performed in a regular manner, generating

homogeneous regions without incomplete residual areas.

Once the image has been adjusted, a fixed size is defined for the subimages (tiles), along with a displacement between consecutive regions that depends on the specified overlap ratio. This overlap allows the same region of the original image to be analyzed from different spatial contexts, reducing the risk of objects being fragmented at tile boundaries.

Next, the image is systematically traversed using a sliding window that extracts subimages of constant size. Each tile is processed independently and stored together with its corresponding annotations, which are adapted to the local coordinate system of the subimage. Only annotations that intersect with the extracted region are retained, while those that do not provide relevant information for the local context are discarded.

The final outcome of the algorithm is a new set of images and annotations that can be directly used for training or evaluating a standard YOLO detector. By operating on more localized regions, small objects shall occupy a larger proportion of the analyzed area, which makes their representation more effective during the detection process without introducing modifications to the model architecture.

As a concrete illustration, consider an AirBirds image at its native resolution of 1920×1080 pixels with $T = 640$ and $r = 0.25$, with a displacement of $s = T - r \cdot T = 480$ pixels. Along the horizontal

Algorithm 1. Comparison of AP50 results between the previous work and the proposed approach

Require: Original image I of size (W, H) , tile size T , overlap ratio r .
Ensure: Set of subimages T and adapted annotations L .

```

1   Compute the adjusted size  $(W', H')$  of  $I$  to allow regular partitioning
2   Resize  $I$  to  $(W', H')$  if necessary
3   Compute the displacement  $s = T - r \cdot T$ 
4   Initialize empty sets  $T$  and  $L$ 
5   for  $y = 0$  to  $H' - T$ , with step  $s$  do
6       for  $x = 0$  to  $W' - T$ , with step  $s$  do
7           Extract tile  $I_{x,y}$  of size  $T \times T$  at position  $(x, y)$ 
8           Add  $I_{x,y}$  to  $T$ 
9           Adapt the original annotations that intersect  $I_{x,y}$ 
10          Store the adapted annotations in  $L$ 
11       end for
12   end for
13   return  $T, L$ 

```

axis, the first tile is placed at position 0, and the subsequent tiles are placed at steps of $s = 480$ pixels; since $1920 - 640 = 1280$ and $\lfloor 1280/480 \rfloor = 2$, two additional tiles fit, which means 3 tiles horizontally and an adjusted width of 1600 pixels. Along the vertical axis, $1080 - 640 = 440 < 480$, so no additional tile fits beyond the first one, yielding one tile vertically and an adjusted height of 640 pixels. The algorithm therefore produces 3 tiles in total for this image. Each tile retains only the annotations intersecting its region, with its coordinates transformed in relation to the local tile reference frame; the annotations whose visible area falls below a minimum threshold after clipping are discarded.

4. Experimental Setup

4.1 Dataset

For the experimental evaluation of the proposed approach, the AirBirds dataset (Sun et al., 2022) was used. This public dataset was originally designed for bird detection tasks in highly complex real-world scenarios. The images were captured in airport environments with the aim of preventing bird-aircraft collisions, making the dataset particularly challenging from the perspective of small object detection.

The images have a native resolution of 1920×1080 pixels and cover a wide range of environmental conditions, including different seasons of the year, lighting variations, changing weather conditions, and multiple bird species. One of the most relevant characteristics of the dataset is the reduced size of the annotated instances, as birds often appear at large distances from the camera and occupy a very small fraction of the total image area. In many cases, the bounding boxes associated with the objects have dimensions on the order of only a few pixels, which poses a significant challenge for object detection models.

Although the dataset includes official training and test splits, inconsistencies were identified in the annotation of the test set in the preliminary study of García, Villar & Sedano (2026), such as bounding boxes which were partially outside the image boundaries. For this reason, and in order to ensure a reliable and consistent evaluation,

this partition was discarded and only the original training set of the dataset was used.

For this set, a random splitting into three subsets was performed: training, validation, and test. Specifically, 70,000 images were selected for training the models, 10,000 images for validation, and 18,141 images for testing. This split of the dataset follows a standard 70/10/20 distribution for the training, validation, and test sets, respectively. This partition was kept constant across all the experiments which were carried out to ensure the comparability of results among the different evaluated configurations.

This distribution follows the widely adopted 80/20 convention for training+validation versus final evaluation, which is a standard one in the field when working with datasets of this scale. Reserving 20% of the data (18,141 images) for the test set ensures a sufficiently large and representative sample for assessing generalization to real-world conditions, while the remaining 80% is further divided into 70% for training and 10% for validation. Although 10% may appear modest as a proportion, it corresponds to 10,000 images in absolute terms - a substantial validation set given the overall size of the dataset - which is adequate for guiding hyperparameter and model selection without excessively reducing the data available for training. This configuration therefore represents a balanced compromise between maximizing the training data and preserving reliable subsets for validation and final testing.

4.2 Models and Configuration

In this work, three recent versions of the YOLO family are employed: YOLOv8 (Jocher, Chaurasia & Qiu, 2023), YOLOv9 (Wang, Yeh & Liao, 2024), and YOLOv11 (Jocher & Qiu, 2024). The selection of these models follows a criterion of continuity and comparability, as they are the same detectors used in the previous study that motivated this work (García, Villar & Sedano, 2026). This choice allows for a direct analysis of the impact of the proposed strategy without introducing architectural variations.

YOLOv8 represents a step forward in the evolution of single-stage detectors, incorporating improvements in the definition of the detection

head and in the training process, with a design aimed at simplifying the YOLO model architecture and enhancing learning stability. In contrast, YOLOv9 introduces modifications oriented toward a more efficient management of information during forward propagation, with the goal of improving the generalization capability and the utilization of features extracted by the network.

YOLOv11, as the most recent version, continues this evolutionary line by incorporating additional adjustments in the architecture and prediction mechanisms of YOLO, seeking a better balance between accuracy and computational efficiency. Despite these differences, the three models share a fundamental characteristic: they perform object detection and classification simultaneously over feature maps generated from an image resized to a fixed input size.

For each of the YOLO model versions considered in this study, the two smallest variants were selected, namely the n/t and s versions: YOLOv8n, YOLOv8s, YOLOv9t, YOLOv9s, YOLOv11n, and YOLOv11s (Adarsh, Rathi & Kumar, 2020)..

With regard to the tiling strategy applied during training, the images were divided into 640×640 pixel tiles, using a 25% overlap between adjacent tiles. This strategy helped mitigate information loss at image borders and improved the detection of partially visible objects.

The overlap ratio was empirically determined through preliminary experiments with values in the range typically explored in the literature, where overlap is commonly treated as a dataset- and task-dependent hyperparameter rather than as a fixed value. While time-series and generic sliding-window applications often adopt lower overlap ratios (e.g. around 10%), a 25% overlap was found to provide the best trade-off in this setting, improving the detection of small, boundary-crossing bird instances without excessively increasing the number of generated tiles and the associated computational cost. It should also be noted that overlap is applied exclusively between tiles belonging to the same image, after the dataset has already been partitioned into training, validation, and test subsets (subsection 4.1); tiles from different images are therefore never

overlapping, and no information leakage across subsets is introduced by this step.

Table 1 includes the complete set of parameters used during training. These parameters were determined through preliminary experiments with different configurations, and the final selected values were those that provided a greater training stability and led to the best performance in terms of the AP50 metric (Average Precision with 50% overlap).

Table 1. Training parameters used for all YOLO models

Parameter	Value
Tile size	640×640 pixels
Tile overlap	25%
Number of epochs	100
Batch size	16
Optimizer	AdamW
Momentum	0.9
Initial learning rate	0.0001

4.3 Metrics

For the evaluation of model performance, the Average Precision at an Intersection over Union threshold of 50% (AP50) was selected as the primary metric. This metric was already employed in the preliminary study and it is widely used in object detection tasks to jointly quantify a detector's localization and classification capabilities.

The AP metric is derived from the precision–recall curve, which summarizes the trade-off between the proportion of correct detections and the model's ability to identify all the instances present in the image. In the case of AP50, a detection is considered correct when the overlap between the predicted bounding box and the ground-truth annotation, measured using the Intersection over Union (IoU) index, is equal to or greater than 50%. This threshold allows the evaluation of whether the model can localize the object with a reasonable accuracy, without requiring an excessively strict match between the prediction and the annotation.

In the preliminary study, more restrictive metrics such as AP75 were also employed, which require a higher localization accuracy of the

bounding boxes. However, the analysis of the results showed that both metrics exhibited a very similar behavior and did not lead to substantial differences in the comparison between models. In particular, the conclusions drawn with regard to AP50 and AP75 were consistent, suggesting that the observed limitations were primarily related to object detection capability rather than to fine-grained localization accuracy.

For this reason, in this work AP50 was exclusively selected as a comparison metric. This choice allows the analysis to focus on the main objective of the study, which is to evaluate the ability of the employed models to detect small objects in high-resolution images and to analyze the impact of image partitioning on this capability. In addition, the use of a single metric facilitates result interpretation and the direct comparison among the different evaluated configurations.

To account for training variability, each model configuration was trained three times. The results are reported as the mean $\pm$ standard deviation (St. Dev.) across the three runs.

5. Results and Discussion

Table 2 summarizes the AP50 results obtained by the evaluated YOLO model versions under the proposed training configuration.

Notably, all model configurations produced similar results across the three independent training runs yielding a standard deviation with low values for all reported AP50 values. This indicates that the training pipeline exhibits a limited variability across the training runs; however, given the magnitude of these standard deviations relative to the small differences observed between some model variants (see below), certain comparisons should be interpreted with caution rather than as robust architectural effects.

Overall, the results show a consistent performance across the different model families, with AP50 mean values ranging from 0.2628 to 0.2888. Among all the evaluated architectures, YOLOv9s achieved the highest performance, with a mean of AP50 of 0.2888. This suggests that, under the proposed tiling strategy and training configuration, the small variant of YOLOv9 provides the best balance between the representational capacity and generalization performance.

When comparing model sizes for each model version, different behaviors can be observed. For YOLOv8, the s variant slightly outperforms the n variant (0.2818 vs. 0.2812), indicating that the additional model capacity contributes positively to the detection performance, although the improvement is modest. A similar trend is observed for YOLOv9, where the s variant outperforms the t version (0.2888 vs. 0.2628), showing a more noticeable gain when increasing model size.

In contrast, YOLOv11 exhibits a different pattern: the smallest variant (n) achieves a slightly higher AP50 than the s version (0.2742 vs. 0.2720). This suggests that, for this architecture, increasing model size does not necessarily translate into an improved performance under the current experimental setup, possibly due to the optimization dynamics or dataset-specific characteristics.

In general, the performance differences between the smallest and small variants remain moderate, indicating that both configurations are capable of learning discriminative features under the proposed training strategy. For the comparison with the previous work (García, Villar & Sedano, 2026), the smallest model variants were considered, namely the n versions for YOLOv8 and YOLOv11 and the t version for YOLOv9. The quantitative comparison of the results reported

Table 2. Detection results obtained by the evaluated YOLO models

Model version	Smallest (n/t)		Small (s)	
	Mean	St. Dev.	Mean	St. Dev.
YOLO v8	0.2812	0.0133	0.2818	0.0054
YOLO v9	0.2628	0.0065	0.2888	0.0190
YOLO v11	0.2742	0.0061	0.2720	0.0181

in the previous work and those obtained by the proposed approach is presented in Table 3.

As shown in Table 3, all the evaluated models exhibit a substantial improvement in terms of AP50 for the proposed approach. Specifically, this approach achieves average performance gains of +0.1802, +0.1518, and +0.1842 for YOLOv8n, YOLOv9t, and YOLOv11n, respectively, representing a marked increase in the detection performance in comparison with the results reported by the previous work.

Although the absolute AP50 values are still moderate and do not indicate a perfect detection performance, the observed improvements are highly significant when considering the inherent difficulty of detecting small objects by using YOLO-based architectures. YOLO models are known to experience performance degradation when dealing with small-scale targets, particularly in high-resolution scenes where object sizes occupy only a limited number of pixels.

In this context, the tiling strategy proposed in this work plays a key role in enhancing the object detection capabilities. By dividing the input images into overlapping tiles, the effective resolution of small objects is increased during training and inference, enabling the models to better capture fine-grained spatial details. The consistent improvements observed across all the evaluated model families suggest that the proposed tiling approach effectively mitigates one of the main limitations of YOLO-based detectors, significantly improving their ability to detect small objects in challenging scenarios.

The general analysis of the results shows that the proposed approach remains model-agnostic and can be seamlessly integrated into the existing YOLO-based detection pipelines.

The experimental results obtained on the AirBirds dataset demonstrate that the proposed

tiling strategy leads to substantial performance improvements across all the evaluated YOLO versions. In comparison with the results reported in the previous work, the smallest model variants achieved consistent average gains of more than +0.15 for AP50, representing a significant enhancement in the small object detection capability. These findings confirm that an appropriate preprocessing strategy can effectively alleviate the limitations imposed by global resizing and significantly improve the performance of YOLO-based detectors in challenging small object detection scenarios.

6. Conclusion

This work has addressed a well-known structural limitation of YOLO-based single-stage detectors when applied to scenarios dominated by small objects in high-resolution images. In particular, the adverse effect of global input image resizing on the spatial representation of small targets was identified as a key factor contributing to the reduced detection performance observed in previous studies.

To mitigate this limitation, a region-based preprocessing strategy based on image tiling was proposed. The method partitions the original image into overlapping local regions that are processed independently by the detector, allowing small objects to be analyzed at a higher effective resolution without modifying the network architecture or altering the standard training procedure.

Several directions for future research arise from the findings presented in this work. First, the proposed approach could be evaluated using larger and more diverse datasets than the AirBirds dataset employed in this study. While the current results demonstrate that the tiling strategy significantly improves small object detection performance under limited data conditions, further

Table 3. Comparison of AP50 results between the previous work and the proposed approach

Model version	Previous work	Smallest (n/t)	Small (s)
YOLO v8	0.101	0.2812 (+0.1802)	0.2818 (+0.1808)
YOLO v9	0.111	0.2628 (+0.1518)	0.2888 (+0.1778)
YOLO v11	0.090	0.2742 (+0.1842)	0.2720 (+0.1820)

experimentation with larger-scale datasets would allow a deeper analysis of the interaction between data availability and preprocessing strategies. In particular, it would be of interest to assess whether the observed performance gains remain consistent, increase, or stabilize as the amount and variability of the training data grow.

Second, an extension of the proposed method toward an adaptive tiling strategy represents a promising research direction. Instead of applying a fixed tile size and overlap uniformly across the entire image, future work could focus on dynamically adjusting these parameters based on the spatial distribution and density of small objects. Regions with a higher density or confluence of small objects could be analyzed at a higher spatial resolution, while areas where small objects are sparse could be processed more coarsely. Such an adaptive approach has the potential to further improve the detection performance while reducing the unnecessary computational overhead.

Finally, the applicability of the proposed tiling strategy could be explored in other computer

vision tasks characterized by the presence of small objects. Domains such as aerial and satellite imagery, traffic surveillance, wildlife monitoring, or industrial inspection share similar challenges related to object scale and spatial sparsity. Evaluating the proposed preprocessing technique across different datasets and application contexts would help validate its generality and reinforce its potential as a practical solution for small object detection beyond the specific scenario addressed in this work.

Acknowledgements

This research has been funded by the Spanish Research Agency - grant no. PID2023-146257OB-I00, by Principado de Asturias, grant no. IDE/2024/000734, and by the Council of Gijón through the University Institute of Industrial Technology of Asturias - grants no. SV-25-GIJÓN-1-23 and SV-26-GIJÓN-1-02. This work was also supported through the ICECyL project (Junta de Castilla y León) under project no. CCTT5/23/BU/0003 (ANASTRA) and CCTT5/23/BU/0005 (SYNCROBOT).

REFERENCES

- Aboyomi, D. D. & Daniel, C. (2023) A Comparative Analysis of Modern Object Detection Algorithms: YOLO vs. SSD vs. Faster R-CNN_2023. *ITEJ (Information Technology Engineering Journals)*. 8(2), 96–106. <https://doi.org/10.24235/itej.v8i2.123>.
- Adarsh, P., Rathi, P. & Kumar, M. (2020) YOLO v3-Tiny: Object Detection and Recognition using one stage improved model. In: *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)*, 6-7 March 2020, Coimbatore, India. New York, USA, IEEE. pp. 687-694. <https://doi.org/10.1109/ICACCS48705.2020.9074315>.
- Benjumea, A., Teeti, I., Cuzzolin, F. et al. (2021) YOLO-Z: Improving small object detection in YOLOv5 for autonomous vehicles. [Preprint] <https://arxiv.org/abs/2112.11798> [Accessed: 22nd January 2026].
- Dicu, M., González, E. G., Chira, C. et al. (2024) The Impact of Data Annotations on the Performance of Object Detection Models in Icon Detection for GUI Images. In: Quintian, H., Corchado, E., Troncoso Lora, E. et al. (eds.) *Hybrid Artificial Intelligent Systems (19th International Conference, HAIS 2024, 9–11 October 2024, Salamanca, Spain, Part I, part of Lecture Notes in Artificial Intelligence series, vol. 14857)*. Berlin, Heidelberg, Springer-Verlag, pp. 251–262. https://doi.org/10.1007/978-3-031-74183-8_21.
- García, E., Villar, J. R. & Sedano, J. (2026) Free-Range Chicken Farms Oriented Bird Detection. In: Chira, C., Matei, O., Pop, F. et al. (eds.) *Innovative Perspectives on Computational Intelligence and Data Science (InnoComp 2025, Communications in Computer and Information Science, vol 2794)*. Cham, Switzerland, Springer Nature, pp. 197–207. https://doi.org/10.1007/978-3-032-12481-4_12.
- Jeong, D. & Kim, J. (2022) Road Damage Detection using YOLO with Image Tiling about Multi-source Images. In: *2022 IEEE International Conference on Big Data (Big Data)*, 17-20 December 2022, Osaka, Japan. New York, USA, IEEE. pp. 6401–6406. <https://doi.org/10.1109/BigData55660.2022.10020282>.
- Jocher, G., Chaurasia, A. & Qiu, J. (2023) *Ultralytics YOLO8*. Version 8.0.0. <https://github.com/ultralytics/ultralytics>. [Accessed: 21st February 2026].
- Jocher, G. & Qiu, J. (2024) *Ultralytics YOLO11*. Version 11.0.0. <https://github.com/ultralytics/ultralytics>. [Accessed: 21st February 2026].
- Klco, P., Koniar, D., Hargas, L. et al. (2023) Quality inspection of specific electronic boards by deep neural networks. *Scientific Reports*. 13(1), Art. No. 20657. <https://doi.org/10.1038/s41598-023-47958-0>.

- Lu, X., Li, Q., Li, B. et al. (2020) MimicDet: Bridging the Gap Between One-Stage and Two-Stage Object Detection. In: Vedaldi, A., Bischof, H., Brox, T. et al. (eds.) *Computer Vision -- ECCV 2020*. (part of the *Lecture Notes in Computer Science* series, vol. 12359). Cham, Switzerland, Springer International Publishing, pp. 541–557.
- Moraes, A. M., Pugliese, L. F., dos Santos, R. F. et al. (2025) Effectiveness of YOLO Architectures in Tree Detection: Impact of Hyperparameter Tuning and SGD, Adam, and AdamW Optimizers. *Standards*. 5(1), 9. <https://doi.org/10.3390/standards5010009>.
- Ren, S., He, K., Girshick, R. et al. (2015) Faster R-CNN: Towards real-time object detection with region proposal networks. In: *NIPS'15: Proceedings of the 29th International Conference on Neural Information Processing Systems (Vol. 1)*, 7-12 December 2015, Montreal, Canada. Cambridge, MA, USA, MIT Press. pp. 91-99.
- Shewajo, F. A. & Fante, K. A. (2023) Tile-based microscopic image processing for malaria screening using a deep learning approach. *BMC Medical Imaging*. 23(1), Art. No. 39. <https://doi.org/10.1186/s12880-023-00993-9>.
- Sun, H., Wang, Y., Cai, X. et al. (2022) AirBirds: A Large-scale Challenging Dataset for Bird Strike Prevention in Real-world Airports. In: *Proceedings of the 16th Asian Conference on Computer Vision (ACCV)*, 4-8 December 2022, Macao, China. New York, USA, Association for Computing Machinery. pp. 2440–2456.
- Wang, C.-Y., Yeh, I.-H. & Liao, H.-Y. M. (2024) YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information. In: Leonardis, A., Ricci, E., Roth, S. et al. (eds.) *Computer Vision – ECCV 2024 (18th European Conference, 29 September–4 October 2024, Milan, Italy, Proceedings, Part XXXI)*, part of *Lecture Notes in Computer Science* series, vol. 15089). Cham, Switzerland, Springer, pp. 1–21. https://doi.org/10.1007/978-3-031-72751-1_1.
- Yu, M. & Leung, H. (2025) Two-Stage Approach to Small-Object Detection. *Systems Engineering*. 28(4), 548–556. <https://doi.org/10.1002/sys.21800>.



This is an open access article distributed under the terms and conditions of the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.