Revista de Informática Teórica e Aplicada - RITA - ISSN 2175-2745
Vol. 33, Num. 2 (2026) 98-105

RESEARCH ARTICLE

Comparative Evaluation of YOLO-family Detectors for Pig Detection in Precision Livestock Systems

Avaliação Comparativa de Detectores da Família YOLO para Detecção de Suínos em Sistemas de Pecuária de Precisão

Marcos Vinicius Mendes Faria[1]*, Thiago Meireles Paixão[1], Francisco de Assis Boldt[1]

Abstract: Precision Livestock Farming (PLF) delivers technological solutions for modern pork production, where real-time monitoring is vital for animal health and welfare. Object detection models are key in PLF, yet comprehensive baselines for recent architectures on specialized datasets remain limited. This study benchmarks two state-of-the-art models, YOLOv8 and YOLOv12, across five sizes (nano to extra-large) for pig detection using the public PigLife dataset. Ten models were fine-tuned and evaluated with standard COCO metrics. Results show that while accuracy improves with model size, gains diminish with larger models. The YOLOv8m, with 25.9 million parameters, achieved a near-peak mean Average Precision (mAP) of 91.8%. This work establishes a new benchmark on PigLife, surpassing previous results, and highlights the trade-off between YOLOv8's higher accuracy and YOLOv12's parameter efficiency.
Keywords: precision livestock farming — object detection — deep learning — YOLOv8 — YOLOv12 — pig monitoring — computer vision

Resumo: A Pecuária de Precisão (PLF) oferece soluções tecnológicas para a produção moderna de suínos, onde o monitoramento em tempo real é essencial para a saúde e o bem-estar animal. Modelos de detecção de objetos são fundamentais na PLF, mas ainda há poucas referências abrangentes para arquiteturas recentes em conjuntos de dados especializados. Este estudo avalia dois modelos de ponta, YOLOv8 e YOLOv12, em cinco tamanhos (do nano ao extra-large) para detecção de suínos usando o conjunto público PigLife. Dez modelos foram ajustados e avaliados com métricas padrão COCO. Os resultados mostram que, embora a acurácia aumente com o tamanho do modelo, os ganhos diminuem nos maiores. O YOLOv8m, com 25,9 milhões de parâmetros, atingiu uma mAP próxima do pico de 91,8%. O trabalho estabelece um novo benchmark no PigLife e evidencia o equilíbrio entre a maior acurácia do YOLOv8 e a eficiência do YOLOv12.
Palavras-Chave: pecuária de precisão — detecção de objetos — deep learning — YOLOv8 — YOLOv12 — monitoramento de suínos — visão computacional

[1] *Instituto Federal de Educação, Ciência e Tecnologia do Espírito Santo (IFES), Brazil*
***Corresponding authors**: marcosfaria03@hotmail.com, {thiago.paixao, franciscoa}@ifes.edu.br
DOI: http://dx.doi.org/10.22456/2175-2745.150940 • **Received:** 14/10/2025 • **Accepted:** 03/11/2025

1. Introduction

Pork plays a crucial role in global food security and the agricultural economy, being one of the most consumed animal proteins worldwide. The meat market has shown significant growth: according to the Meat Market Review produced by the Food and Agriculture Organization of the United Nations (FAO), the price index reached 117.3 points in 2024, representing an increase of 2.8% compared to 2023. In the same year, global meat production expanded by 1.7%, reaching 379 million tonnes, with pork production remaining stable during this period [1]. Projections by the FAO also indicate that global meat production will need to increase substantially to meet the demands of a growing world population, with overall meat demand projected to reach 470 million tonnes by 2050, an increase of more than 200 million tonnes compared to levels in 2005 and 2007 [2].

This increasing demand scenario imposes complex challenges on pork producers. The rise in the number of animals per farm, driven by the need for greater production with a declining rural workforce, makes individual animal monitoring increasingly difficult. Issues such as large-scale animal health and welfare, reducing the environmental impact of livestock farming, and ensuring profitability become essential. In this context, Precision Livestock Farming (PLF) emerges as a promising solution for sustainable livestock production. PLF involves the use of technologies, such as cameras, sensors, and acoustic devices, integrated with artificial intelligence, for continuous and real-time monitoring of the herd's production, reproduction, health, welfare, and environmental impact [3].

In the context of PLF, the field of Computer Vision (CV) has proven to be a valuable tool for animal monitoring, especially in tracking the productive cycle of swine. Notable examples include the work proposed in [4], which introduces a real-time, non-intrusive approach to identifying and tracking pig movements in farm environments, thus contributing to obtaining essential information for animal health.

Although promising for optimizing pig farming, CV faces significant barriers to widespread commercial use. A study conducted by researchers at the University of Illinois Urbana-Champaign [5] aimed to identify the main challenges in applying computer vision systems in pig production facilities. One of the primary challenges identified was the inherent complexity of monitoring animal behavior. The precise identification of behavioral changes represents a time-consuming task even for caregivers in large commercial swine production systems, requiring accurate observation skills and a deep practical knowledge of animal management and livestock system operation.

Since object detection is a fundamental step in computer vision for PLF, this study benchmarks the state-of-the-art models YOLOv8 [6] and YOLOv12 [7] in different sizes. This analysis establishes a performance baseline for these pre-trained and fine-tuned models to guide future work. The motivation of this study is to investigate the differences among YOLO-family models, focusing particularly on the impact of their different sizes. The goal is to identify an optimal option that balances performance and efficiency, thereby promoting cost reduction and accessibility in Precision Livestock Systems, specifically within the context of pig detection.

The experimental results reveal that YOLOv8 delivers, in most cases, higher accuracy than YOLOv12, while YOLOv12 stands out for its superior parameter efficiency, making it more suitable for edge devices with limited resources. Among the evaluated models, YOLOv8m achieved the best balance between performance and complexity, setting a new benchmark with an mAP of 91.8% on the PigLife dataset [8]. These findings highlight the trade-offs between accuracy and efficiency, offering valuable guidance for practical deployments in Precision Livestock Systems.

The rest of this work is organized as follows. Section 2 reviews related studies, highlighting existing approaches and the main challenges addressed in the literature. Section 3 describes the materials and methods, including the dataset used, the fine-tuned models evaluated, the experimental setup, the evaluation metrics, and the computational resources. Section 4 presents and discusses the obtained results. Section 5 summarizes the conclusions of this study and outlines directions for future work.

2. Related Work

The application of computer vision and deep learning to automate swine monitoring is a rapidly growing field, aiming to enhance farm management, animal welfare, and productivity. Research in this area addresses a wide range of tasks, from analyzing complex behaviors to solving fundamental challenges in object detection. Deep learning models, particularly architectures based on Convolutional Neural Networks (CNNs), have become the standard approach for these tasks. For instance, models have been developed to classify specific pig behaviors such as aggression [9] or to monitor health indicators like posture and drinking habits [10], demonstrating the potential for early disease detection.

Despite these advances, a primary challenge is accurate pig detection in realistic, often crowded, and visually complex farm environments. Overcoming issues like occlusion and tracking small, young pigs is a central theme in recent literature. To address this, some researchers have focused on improving existing object detection frameworks. Peng et al. [11], for example, enhanced the YOLOv5 model by integrating an attention module and an extra detection head to better identify small pigs, achieving a notable increase in mAP from 67.6% to 76.4%. In cases of extreme density where individual detection is impractical, alternative methods like density map estimation have been proposed. Feng et al. [12] developed a model to count and localize pigs by predicting a density map from an image, offering a robust solution for congested scenes. Beyond model architecture, a critical barrier to deployment is the high cost of data labeling and the domain shift problem, where models trained in one environment fail in another. Lee et al. [13] tackled this by creating a self-training system that adapts a model to new conditions with minimal labeled data, significantly improving performance.

Based on the work of He et al. [14], the PDC-YOLO network was developed to address the challenge of automated pig detection and counting in complex farming environments. Building upon YOLOv7, their approach integrates SPD-Conv modules to enhance feature extraction from low-resolution images and small objects, and replaces the original neck with an Asymptotic Feature Pyramid Network (AFPN) to better fuse multi-scale features. Furthermore, they employ rotated bounding boxes to improve localization accuracy for variably oriented pigs. Evaluated on a diverse dataset containing various pig types, lighting conditions, and perspectives, PDC-YOLO achieved a mAP@0.5 of 91.97%, demonstrating superior performance compared to other YOLO variants and significantly outperforming manual counting in speed with a low error rate.

A foundational challenge tying all this research together is the need for standardized, high-quality public datasets that reflect real-world commercial farm conditions. To address this gap, Li et al. [8] introduced PigLife, a large-scale dataset featuring over 22,000 instances with significant visual difficulties. They established comprehensive benchmarks by evaluating a wide array of models, from those trained from scratch to fine-tuned and zero-shot approaches. Their findings revealed that fine-tuned models significantly outperformed others, with YOLOv8m achieving the highest performance with an Average Precision (AP) of 91.3.

Our study builds directly upon the work of Li et al. [8], similar to their approach, we utilize the PigLife dataset to conduct an analysis of fine-tuned object detection models. However, our study distinguishes itself by focusing specifically on a systematic comparison of YOLO-family models. We investigate the trade-offs between model size and accuracy by evaluating not only YOLOv8 but also the more recent YOLOv12, benchmarking all available model sizes (nano, small, medium, large, and extra-large), providing a comparative analysis of YOLOv8 and YOLOv12 variants for pig detection, offering insights into how the number of parameters affects precision.

3. Methods

This section details the experimental pipeline, beginning with the description of the PigLife dataset, followed by the characteristics of the pre-trained models adopted, the training and validation strategy, and the evaluation metrics employed. Finally, the computational resources used to conduct the experiments are presented. Together, these methodological steps establish a consistent framework for comparing YOLOv8 and YOLOv12 across different model sizes.

3.1 Dataset

As previously mentioned, the dataset used will be PigLife, developed by the AIFARMS[1] institute at the University of Illinois. This dataset covers seven phases of the swine production cycle: breeding, gestation, farrow, wean, nursery, growth, and finish, as we can see in Figure 1. All videos from PigLife were extracted from continuous recordings made by surveillance cameras installed in swine farms. All clips were processed to remove backgrounds containing people or irrelevant information. From each video, images were generated at 1-second intervals, totaling 2,130 images and 21,869 pig instances. Each instance was annotated with segmentation masks by a group of specialists and validated by another group. The images were named following a structure composed of five parts: growth stage (SID, $1000 \sim 1100$), image description (IID, $1101 \sim 2000$), housing condition (EID, $2001 \sim 5000$), animal description (AID, $5001 \sim 6000$), and frame number (FN, $0 \sim \infty$) [8]. The next subsection introduces the pre-trained YOLO-family models selected for fine-tuning.

3.2 Pre-trained models

For this work, we use 10 variants of the YOLO-family models, all pre-trained on the COCO dataset [15]. First, we selected the latest version of the YOLO model, YOLOv12, and fine-tuned it on the PigLife dataset in all available sizes: n (nano), s (small), m (medium), l (large), and x (extra-large). The same approach was applied to YOLOv8, a well-established and widely adopted version.

YOLOv8 is introduced as a versatile, real-time object detector that builds upon the strong foundation of previous YOLO iterations while integrating significant architectural and methodological innovations. Its design combines a refined CSPNet backbone for enhanced feature extraction, an optimized FPN+PAN neck for superior multi-scale object detection, and an anchor-free bounding box prediction approach that simplifies training and improves adaptability. Additionally, YOLOv8 incorporates advanced training techniques, such as mosaic and mixup augmentation, focal loss, and mixed-precision training, boosting both accuracy and efficiency. These innovations make YOLOv8 suitable for a wide range of applications, from edge deployments to high-precision industrial tasks. It was selected for this project due to its well-established reputation and proven performance, offering a balance of speed, accuracy, and scalability across its range of model sizes. Additionally, the study by Li et al. [8] demonstrates the viability of YOLOv8 using only the medium-sized YOLOv8m model, which has 25.9 million parameters. Our study complements their investigation by evaluating all available variants to compare how their different sizes impact the mean Average Precision.

YOLOv12 is presented as a pioneering attention-centric, real-time object detector designed to overcome the traditional trade-off between the high performance of attention mechanisms and the high speed of CNN-based models. Its architecture is distinguished by two primary innovations: a simple yet efficient Area Attention (A2) module and Residual Efficient Layer Aggregation Networks (R-ELAN). The A2 module reduces the computational complexity of attention while maintaining a large receptive field, while R-ELAN introduces block-level residual connections to solve optimization challenges that arise when integrating attention mechanisms, especially in larger models. YOLOv12 was selected for this study because it demonstrably surpasses all popular real-time object detectors in accuracy with competitive speed, achieving superior accuracy at a competitive speed; for instance, its smallest version outperforms YOLOv10-N by 2.1% mAP while being faster [6].

By leveraging both YOLOv8 and YOLOv12 across all available sizes, we enable a systematic comparison between two state-of-the-art families. The following subsection describes the experimental design adopted to fine-tune and evaluate these models.

3.3 Experiments

For a significant part of the experiments, we utilized the Ultralytics[2] framework on Python to streamline the workflow. The initial preprocessing step involved converting the dataset labels from the original COCO format to the YOLO format. This conversion was performed using the JSON2YOLO[3] util-

[1] https://aifarms.illinois.edu/

[2] https://docs.ultralytics.com/models/

[3] https://github.com/ultralytics/JSON2YOLO

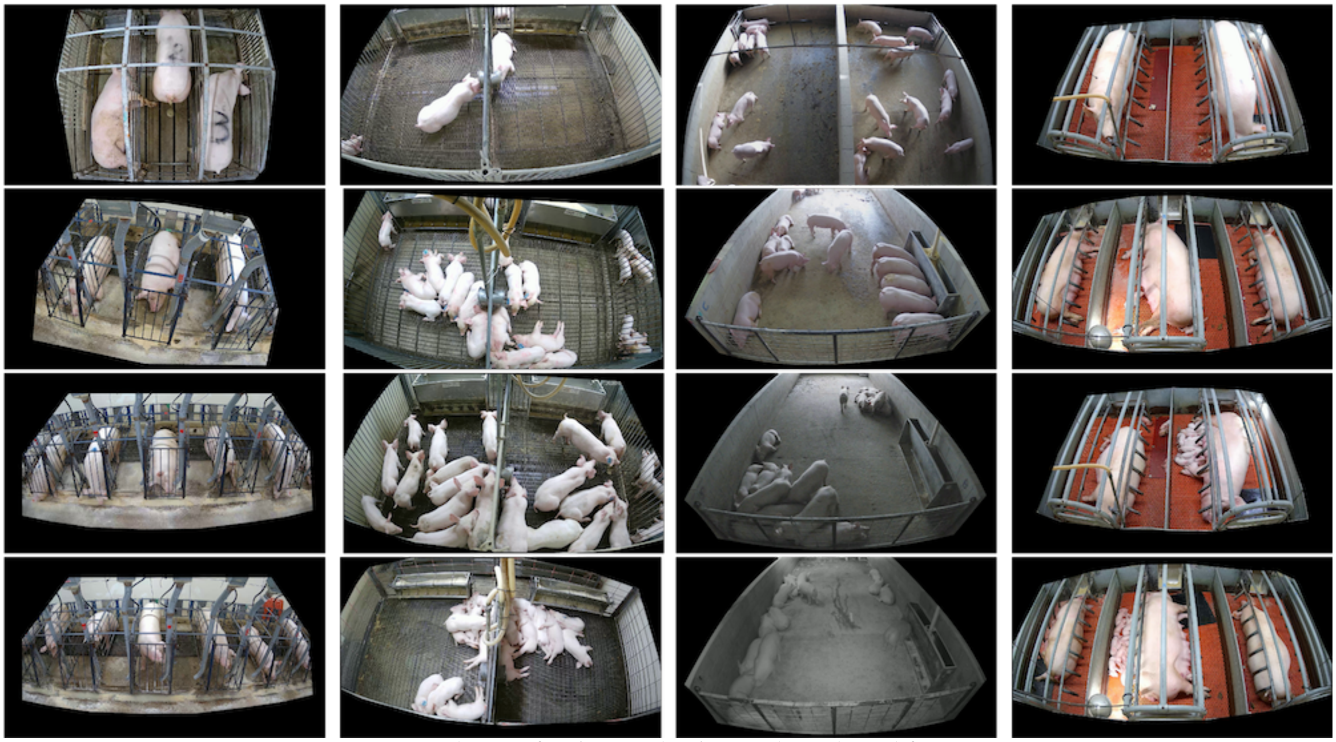


Figure 1. Phases of swine production cycle. Adapted from [8].

ity, also provided by Ultralytics. The original dataset was already partitioned into an 80% training set and a 20% test set. Since the baseline paper [8] defined only the train and test split, our methodology further subdivides the training set, randomly using a simple Python script, to create a dedicated validation set. Resulting in the final data distribution detailed in Table 1.

Table 1. Dataset split for the PigLife dataset into training, validation, and test sets.

Split	Training	Validation	Test
Percentage	65%	15%	20%
Number of Samples	1364	340	426

Following data preparation, we selected all the pre-trained models for YOLOv12 and YOLOv8, ranging from nano to extra-large sizes. The training process for each model was configured to run for 100 epochs. Since the baseline study only specified the number of batches, we followed Ultralytics recommended configuration for the other parameters. All input images were uniformly resized to 640×640 pixels. The models processed images in batches of 4, with two worker threads.

To fine-tune these models, we built an automatic pipeline that loads each pre-downloaded model, fine-tunes it using the split dataset, and, once training is complete, selects the best-performing model, based on mAP, to generate predictions on the test split. This process was repeated until predictions were generated for all models. To ensure the reliability and stability of our results, this entire pipeline was executed six times for each model.

Once all predictions from each model evaluation were obtained, we proceeded to calculate the metrics. For this task, we used the pycocotools library, providing both the predictions file and the ground truth test file. The library then compares the model's predictions against the ground truth to generate the evaluation metrics. The presented results are the average of the six runs; the standard deviation was omitted as it was consistently close to zero, confirming the stability of the outcomes. These metrics were then used to populate Table 2 and to plot the chart shown in Figure 2.

This experimental setup ensures that results are consistent, reproducible, and comparable across all evaluated models. With the training and inference completed, the next step involves defining the evaluation metrics used to assess model performance.

3.4 Evaluation Metrics

For comparative purposes, we adopted the performance metrics used by Li et al. [8]. The performance of all models is evaluated using the standard metrics from the COCO dataset [1], which are based on the concepts of Precision, Recall, and Intersection over Union (IoU).

The IoU measures the overlap between a predicted bounding box and a ground-truth bounding box. The primary metric for object detection is mAP, defined in Equation (1), which is the average of the Average Precision (AP) scores across all N object classes. The AP for a single class is calculated as the area under the precision-recall curve, which provides a combined measure of model quality and completeness. Formally, it is defined as:

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \tag{1}$$

Following the COCO standard, we report the primary AP metric (averaged over IoU thresholds from 0.50 to 0.95), the AP at specific thresholds of 0.50 (AP_{50}) and 0.75 (AP_{75}), and the AP for medium (AP_M) and large (AP_L) objects.

These metrics, aligned with COCO standards, allow for a detailed and reliable assessment of detection accuracy across multiple dimensions. The subsequent subsection outlines the computational resources employed to support the experiments.

3.5 Computational Resources

All experiments were conducted on a single local PC equipped with an Intel Core i9-10900KF 3.70GHz CPU, 32GB of RAM, and an NVIDIA GeForce RTX 3060 GPU with 12GB of dedicated memory. The described hardware and software environment guarantees that the reported results are directly linked to the available computational capacity. With the methodology established, the following section presents and discusses the experimental results.

4. Results and Discussion

Table 2 presents the results generated from the predictions on the test set for the 10 fine-tuned models. This table is divided into three sections: the first section lists the results for YOLOv8 across all five sizes; the second applies the same logic for YOLOv12; and the final row shows the results from Li et al. [8], which we use as a baseline.

Analyzing the direct comparison between the two model families highlights a trade-off between accuracy and parameter efficiency. Across all five model sizes, YOLOv8 consistently achieved higher mAP when compared with YOLOv12 models. For example, the YOLOv8m model reached an mAP of 91.8%, slightly surpassing the YOLOv12m at 91.1%. However, this gain in accuracy comes with a larger model size. As outlined in the methodology, YOLOv12 is an attention-centric detector designed for efficiency, which is evident from its consistently lower parameter count across all size tiers. In particular, YOLOv12m proves competitive in the mid-range, achieving almost the same accuracy as YOLOv8m but with 22% fewer parameters, and YOLOv12x narrows the gap with YOLOv8x to only 0.6 p.p. in mAP while requiring significantly fewer parameters (59.1M vs. 68.2M). These findings suggest that while YOLOv8 dominates in accuracy, YOLOv12 delivers more favorable trade-offs in scenarios constrained by computational resources.

The baseline study [8] reported a mAP of 91.3% using a fine-tuned YOLOv8m model. Furthermore, our larger YOLOv8l and YOLOv8x models improved this score slightly to 91.9%. The observed difference in performance can potentially be attributed to our specific experimental methodology, which is distinguished by the use of a dedicated validation set (a 65% training, 15% validation, and 20% testing split). However, it is important to state that without additional testing over multiple runs, it is not possible to confirm if this performance difference is statistically significant.

A more granular analysis of the results, moving beyond the primary mAP metric, reveals important nuances in model performance, particularly regarding localization precision. The COCO metrics AP_{50} and AP_{75} evaluate performance at "loose" (50% IoU) and "strict" (75% IoU) overlap thresholds, respectively. Our findings in Table 2 show that nearly all tested models achieved AP_{50} scores of 98.9%, indicating that every architecture is highly proficient at the fundamental task of locating pigs within a frame. This saturation suggests that future progress should focus on stricter localization metrics such as AP_{75}. Indeed, the AP_{75} metric exposed more significant performance differences: the YOLOv8m model distinguished itself with a top-tier score of 97.8%, approximately 1 p.p. higher than the others, demonstrating its superior ability to generate precise detections. This observation aligns with Peng et al. [11], who also emphasized the importance of improving detection precision for small or partially occluded pigs through architectural modifications. Such findings highlight that even marginal improvements in AP_{75} can have meaningful implications for real-world monitoring.

Furthermore, examining the models' performance across different object scales using the AP_M (medium objects) and AP_L (large objects) metrics highlights the practical challenges of on-farm monitoring. A consistent trend across all models was the significantly higher performance on large objects compared to medium ones, with AP_L scores consistently above 88% while AP_M scores were considerably lower. This discrepancy underscores the inherent difficulty of detecting pigs that are distant from the camera, partially occluded, or clustered in dense groups, a key barrier in real-world farm environments. Interestingly, while the YOLOv8m excelled in overall metrics, the slightly larger YOLOv8l model achieved the highest performance on medium-sized objects with an AP_M of 76.8%. On the other hand, YOLOv12 models showed more balanced behavior across medium and large object categories, suggesting better robustness when object scales vary substantially. This stability may be advantageous for deployment in farms where the visual scale of pigs fluctuates frequently depending on pen layout and growth phase. These results build directly upon the findings of Li et al. [8], who also identified medium-object detection as a critical bottleneck for swine monitoring, and further demonstrate how architectural choices can shift performance across object scales.

Figure 2 further illustrates these findings by plotting model accuracy against parameter count. The first behavior observable is that mAP increases consistently as we move from the nano (n) to the extra-large (x) models. However, the most significant performance gains occur when transitioning from small to medium-sized models, particularly evident in the YOLOv8 family: YOLOv8n, with 3.2 million parameters,

Table 2. Object detection performance of YOLOv8 and YOLOv12 models across various sizes.

Model	Parameters (M)	*mAP*	AP_{50}	AP_{75}	AP_M	AP_L
YOLOv8n	3.2	88.0	**98.9**	95.7	67.4	88.9
YOLOv8s	11.2	90.6	**98.9**	96.8	72.2	91.5
YOLOv8m	25.9	91.8	**98.9**	**97.8**	75.6	92.4
YOLOv8l	43.7	**91.9**	**98.9**	96.8	**76.8**	**92.6**
YOLOv8x	68.2	**91.9**	**98.9**	96.8	75.1	**92.6**
YOLOv12n	2.6	87.8	98.8	95.7	62.2	88.8
YOLOv12s	9.3	90.2	**98.9**	96.9	71.3	91.0
YOLOv12m	20.2	91.1	**98.9**	96.8	72.9	91.8
YOLOv12l	26.4	91.2	**98.9**	96.7	74.6	92.0
YOLOv12x	59.1	91.3	**98.9**	96.7	73.0	92.2
Li et. al [12] (YOLOv8m)	25.9	91.3	98.0	96.9	77.2	92.0

achieved an mAP of 88.0, whereas YOLOv8m, with 22.7 million additional parameters, yielded an improvement of 3.8 p.p. In contrast, moving from the medium to the extra-large model adds 42.3 million more parameters but results in only a 0.1 p.p. increase in mAP. This plateau behavior indicates that, beyond a certain point, adding parameters contributes minimally to accuracy, underscoring YOLOv8m as the most balanced option between accuracy and computational cost.

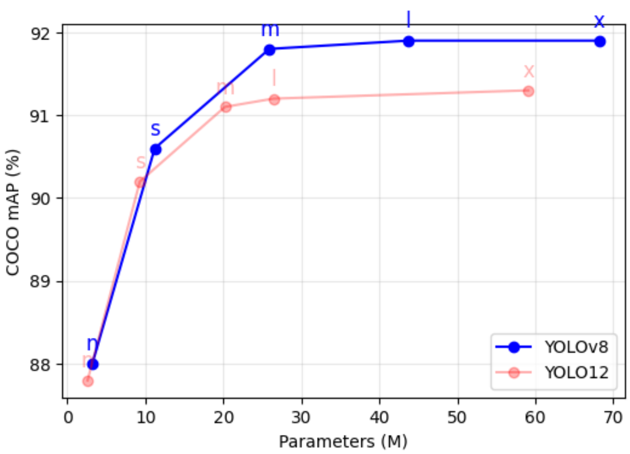


Figure 2. Performance (COCO mAP) of YOLOv8 and YOLO12 models against their size in millions of parameters.

A similar pattern can be seen in the YOLOv12 family, although the slope of improvement differs slightly. YOLOv12n to YOLOv12m shows consistent and meaningful gains (from 87.8% to 91.1%), but beyond the medium model, performance stabilizes, with YOLOv12l and YOLOv12x offering only marginal increases. Notably, YOLOv12 reaches its saturation point earlier than YOLOv8, suggesting that smaller versions of YOLOv12 are already highly competitive in accuracy relative to their size, which strengthens its case for deployment in lightweight or resource-constrained environments.

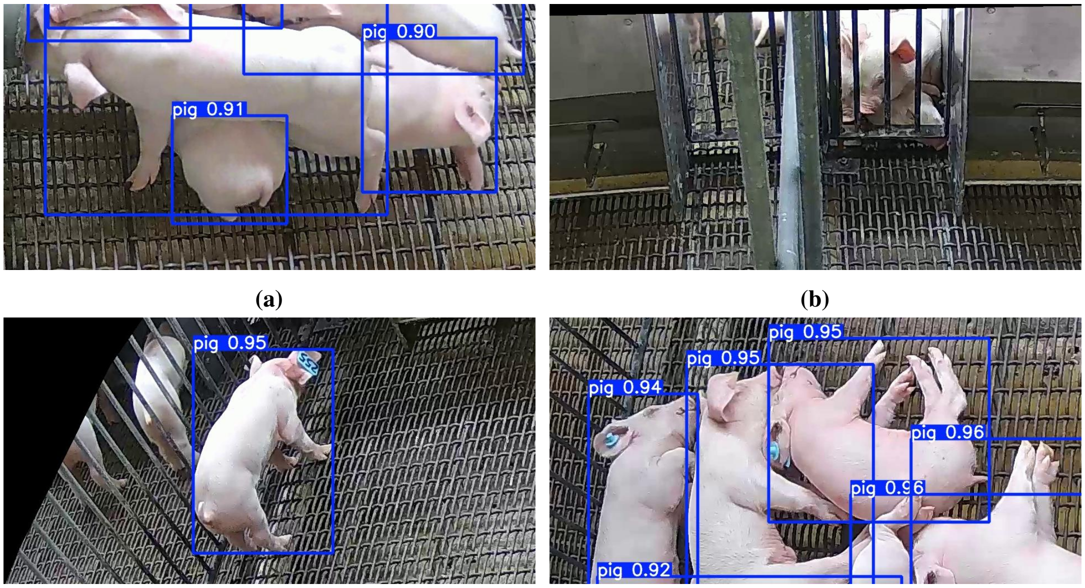
Another important aspect revealed by Figure 2 is the efficiency of YOLOv12 when considering parameter and performance ratios. For instance, YOLOv12m achieves nearly the same accuracy as YOLOv8m, yet with approximately 5.7 million fewer parameters. Similarly, YOLOv12x requires about 9 million fewer parameters than YOLOv8x to approach its accuracy, narrowing the gap to less than 1 p.p. This means that when plotted on the curve, YOLOv12 consistently shifts downward in parameter count without losing substantial accuracy, highlighting its architectural advantage in terms of efficiency.

Finally, when considering the slope of the curve between small and medium sizes, YOLOv8 shows the steepest accuracy gain, making YOLOv8m the "sweet spot" for farms seeking high precision without excessive computational costs. On the other hand, the curve of YOLOv12 is more gradual, showing smoother improvements across model sizes. This smoothness may indicate that YOLOv12 scales more predictably across different deployment scenarios, while YOLOv8 exhibits sharper jumps that must be carefully weighed against resource availability.

To complement the quantitative results presented in Table 2 and Figure 2, Figure 3 provides a qualitative analysis of the YOLOv8m model, illustrating its performance on challenging scenarios from a test set image. These examples visually suggest the model's potential strengths and weaknesses, particularly concerning different types of occlusion.

The model's robustness against pig-to-pig occlusion, a common difficulty in dense farm environments, is suggested in segments 3a and 3c. For instance, in segment 3a, the model successfully identifies one pig with only its rear visible (confidence: 0.91) and another with only its head showing (confidence: 0.90), which may indicate a good capacity to recognize pigs from partial features.

On the other hand, the inference also highlights some of the model's potential limitations, which align with the challenges identified in detecting medium-sized and occluded objects. In segments 3b and 3d, the model failed to detect pigs partially blocked by the metal bars of the enclosure. This observation suggests that while YOLOv8m appears to be robust to biological occlusion, its performance may be compromised by structural or environmental occlusions. These



(a) (b) (c) (d)

Figure 3. Qualitative results from the YOLOv8m model on test image 1010s1121s2001-2s5302-1-301. Segments (a) and (d) show successful detections under pig-to-pig occlusion, while segments (b) and (c) highlight missed detections caused by structural occlusions, such as metal bars.

specific failure cases suggest that future research could focus on improving model resilience to these conditions.

In summary, the combined analysis of Table 2, Figure 2 and Figure 3 reveals that while YOLOv8 consistently reaches higher peak accuracy, YOLOv12 demonstrates superior parameter efficiency and smoother scaling across model sizes. The saturation of AP_{50} across all models suggests that future improvements should prioritize stricter localization AP_{75} and medium-object detection AP_M, which remain the most challenging aspects of pig detection in realistic farm conditions. Moreover, the results demonstrate that YOLOv8m and YOLOv8s are particularly well suited for environments requiring high precision or limited computational resources, respectively, whereas YOLOv12 models stand out as robust alternatives for scenarios where efficiency and scalability are paramount. These insights reinforce that the choice of an optimal detector cannot rely solely on raw accuracy but must balance accuracy, parameter efficiency, and the specific demands of PLF deployments.

Another important insight from our study is that pig localization at lower thresholds AP_{50} appears to be a largely solved problem, with nearly all models surpassing 98.9%. By contrast, medium-object detection AP_M and stricter localization AP_{75} remain challenging, as reflected in the lower AP_M scores across all models. These findings highlight the need for future research to focus on scenarios where pigs appear smaller due to occlusion, crowding, or camera distance. Addressing these challenges is critical for advancing PLF applications, where reliable detection under varying farm conditions directly impacts animal welfare monitoring and production efficiency.

From a practical standpoint, the results provide valuable guidelines for model selection. Farms with sufficient computational resources can benefit from YOLOv8m or YOLOv8l, depending on whether strict localization accuracy or robustness to medium-sized objects is prioritized. Conversely, farms with limited hardware capabilities, such as those deploying edge devices, may find YOLOv8s or the more parameter-efficient YOLOv12 family better suited to their operational constraints. These insights emphasize that choosing an optimal detection architecture requires balancing accuracy, efficiency, and deployment context, rather than relying solely on headline mAP results.

5. Conclusions

This work presented a comparative analysis of two state-of-the-art object detection families, YOLOv8 and YOLOv12, applied to the task of pig detection in commercial farm settings. By systematically fine-tuning and evaluating ten models of varying sizes on a carefully partitioned dataset, we were able to assess not only overall detection accuracy but also performance across localization thresholds, object scales, and parameter efficiency.

The results demonstrated that YOLOv8 achieved higher peak accuracy across all model sizes, with YOLOv8m standing out as the most balanced option in terms of accuracy and computational cost. In particular, YOLOv8m reached a mAP of 91.8% and the highest AP_{75} score of 97.8%, underscoring its suitability for applications that require precise localization, such as automated weight estimation and posture analysis. However, our analysis also revealed that YOLOv12 models, while slightly behind in raw accuracy, offered significant advantages in parameter efficiency and stability across object scales. For example, YOLOv12m and YOLOv12x approached the accuracy of their YOLOv8 counterparts with substantially fewer parameters, making them strong candidates for deployment in resource-constrained environments.

Future work will focus on two main directions. First, additional experiments should be conducted to evaluate statistical robustness by repeating training and testing under varying random seeds and data splits. This will provide stronger evidence for the observed performance differences between YOLOv8 and YOLOv12. Second, integrating temporal information from video streams, rather than analyzing frames independently, may improve medium-object detection by leveraging motion cues and temporal consistency. Beyond these technical improvements, future studies should also explore the integration of detection models with downstream PLF applications, such as automated feeding management, health monitoring, and behavioral analysis, thereby bridging the gap between model performance in controlled experiments and its impact on real-world farm productivity.

In conclusion, this study advances the understanding of trade-offs between accuracy and efficiency in modern object detection applied to livestock farming. By contrasting YOLOv8 and YOLOv12 across multiple dimensions, we provide both methodological insights for the research community and practical guidelines for stakeholders in the livestock sector. Ultimately, these contributions support the development of more efficient and reliable computer vision systems, paving the way for broader adoption of artificial intelligence in PLF.

Author Contributions

Marcos Vinicius Mendes Faria: Conceptualization, Methodology, Software, Investigation, Formal Analysis, and Writing – Original Draft. Thiago Meireles Paixão and Francisco de Assis Boldt: Conceptualization, Supervision, Validation, and Writing – Review & Editing.

References

[1] FAO. *FAO Knowledge Repository*. 2025. Accessed: 2025-08-15. Disponível em: ⟨https://openknowledge.fao.org/items/b553c447-f6b2-4ff8-825e-024767b27760⟩.

[2] Crop Biotech Update. *FAO: The World Needs 70% More Food by 2050*. 2024. Accessed: 2025-08-15. Disponível em: ⟨https://www.isaaa.org/kc/cropbiotechupdate/article/default.asp?ID=4778⟩.

[3] BERCKMANS, D. General introduction to precision livestock farming. *Animal Frontiers*, v. 7, n. 1, p. 6–11, Jan 2017.

[4] TAGOE, S. N. A.; ELMIR, M.; AMANQUAH, N. Monitoring of animal movement using computer vision. In: *2024 IEEE 9th International Conference on Adaptive Science and Technology (ICAST)*. Accra, Ghana: [s.n.], 2024. p. 1–6.

[5] LI, J. et al. Barriers to computer vision applications in pig production facilities. *Computers and Electronics in Agriculture*, v. 200, p. 107227, 2022.

[6] VARGHESE, R.; S, M. Yolov8: A novel object detection algorithm with enhanced performance and robustness. In: . [S.l.: s.n.], 2024. p. 1–6.

[7] TIAN, Y.; YE, Q.; DOERMANN, D. *YOLOv12: Attention-Centric Real-Time Object Detectors*. 2025. Online resource. Disponível em: ⟨https://github.com/sunsmarterjie/YOLOv12⟩.

[8] LI, J. et al. Promote computer vision applications in pig farming scenarios: high-quality dataset, fundamental models, and comparable performance. *Journal of Integrative Agriculture*, 2024.

[9] SOUZA, J. et al. Pig aggression classification using cnn, transformers and recurrent networks. In: *Anais do XIX Workshop de Visão Computacional*. Rio Paranaíba/MG: [s.n.], 2024. p. 1–6.

[10] ALAMEER, A.; KYRIAZAKIS, I.; BACARDIT, J. Automated recognition of postures and drinking behaviour for the detection of compromised health in pigs. *Scientific Reports*, v. 10, p. 13665, 2020.

[11] PENG, N.; LI, F.; LUO, X. Plm-yolov5: Improved yolov5 for pig detection in livestock monitoring. In: *2024 International Conference on Intelligent Robotics and Automatic Control (IRAC)*. [S.l.: s.n.], 2024. p. 619–625.

[12] FENG, W.; WANG, K.; ZHOU, S. An efficient neural network for pig counting and localization by density map estimation. *IEEE Access*, v. 11, p. 81079–81091, 2023.

[13] LEE, J. et al. Sustainable self-training pig detection system with augmented single labeled target data for solving domain shift problem. *Sensors*, v. 25, n. 11, 2025.

[14] HE, P. et al. Pdc-yolo: A network for pig detection under complex conditions for counting purposes. *Agriculture*, v. 14, n. 10, 2024. ISSN 2077-0472. Disponível em: ⟨https://www.mdpi.com/2077-0472/14/10/1807⟩.

[15] LIN, T.-Y. et al. *Microsoft COCO: Common Objects in Context*. 2015. Disponível em: ⟨https://arxiv.org/abs/1405.0312⟩.