ISSN 2007-9737

Perspective-Robust Square Gate Detection Using YOLO for Autonomous Drone Racing

Aldrich A. Cabrera-Ponce[1], L. Oyuki Rojas-Perez[2], Nilda G. Xolo-Tlapanco[2], José Martinez-Carranza[2,⋆]

[1] Benemérita Universidad Autónoma de Puebla,
Faculty of Computer Science, Puebla,
Mexico

[2] Instituto Nacional de Astrofísica, Óptica y Electrónica,
Department of Computational Science, Puebla,
Mexico

carranza@inaoep.mx

Abstract. Gate detection using deep learning methods such as YOLO has become popular in recent years due to aerial robotics competitions, including AlphaPilot, IROS Autonomous Drone Racing, Game of Drones, and the A2RL × DCL. In these competitions, unmanned aerial vehicles (UAVs) have to navigate at high speed through a sequence of gates, where accurate detection is essential for navigation planning and reliable pose estimation. However, conventional YOLO detectors rely on bounding boxes, which become inadequate under perspective variations, such as rotations, oblique viewpoints, or scale changes, leading to poor agreement with the actual gate geometry. To tackle this limitation, we propose a novel perspective-aware approach that employs geometric constraints extracted from the four corners obtained via dual bounding-box detections. The method uses bounding-box detections to reconstruct a polygon representation of the gate, enabling a more accurate description of its shape across varying viewpoints. The results show that our proposed method achieves a mean IoU score of 0.95 across three scenarios compared with traditional YOLO-based methods. Furthermore, our approach sustains real-time performance at 150 fps, keeping it suitable for implementation into autonomous drone racing systems.

Keywords. Gate detection, drone racing, A2RL × DCL.

1 Introduction

Drone racing has gained popularity in recent years, emerging as a competition that leverages advances in computer vision, control, navigation, and autonomy. Many competitions, such as IROS [20], NeurIPS [21], and AlphaPilot [2], involve flying drones at high speed in complex circuits composed of gates, using autonomous systems for fast perception and precise navigation. Recently, A2RL × DCL [1] has furthered the field, bringing international teams to design vision and control algorithms that enable UAVs to complete circuits with many gates at high speed. Besides, the UAV's performance is evaluated in terms of visual perception and control to achieve the fastest time.

Therefore, gate detection plays a key role, as UAVs must detect each gate and pass through it at high speed without collision. To address this challenge, several works have employed convolutional neural networks (CNNs), such as SSD [8] and YOLO [28], to detect gates from single images during the flight. Thus, CNN-based detectors enable robust perception pipelines using monocular cameras, providing practical and efficient solutions for real-time gate detection.

In this way, YOLO has become one of the most widely adopted detectors due to its robustness and efficiency in object detection in images. It performs fast inference suitable for autonomous drone racing, where fast perception is essential for control and navigation. Nonetheless, it relies on bounding boxes to identify objects, which limits its ability to represent gate geometry under perspective variations. As a result, detections may

become imprecise when the gate appears rotated, partially occluded, or viewed from angles, leading to inconsistencies between the predicted bounding box and the current gate structure.

In contrast, we propose an approach to enhance gate detection using dual bounding boxes. In this way, we reconstruct a polygonal representation of the gate's geometry from the bounding box corners. As illustrated in Figure 1, the dual bounding box annotation strategy enables accurate detection of gates in different perspectives, regardless of distance or orientation. Furthermore, the proposed annotations remain compatible with standard YOLO training pipelines, eliminating the need for additional methods, while preserving real-time performance.

In addition, we enhance detection under several conditions by incorporating both simulated and real environment datasets with variations in light, background, and gate appearance, conditions challenging for single bounding box detectors. Also, we use data augmentation by changing the background to reduce the need for manual annotations throughout diverse environments. This combination of efficient annotation and geometric consistency enables a more accurate gate detection for autonomous drone racing. The main contributions of this article can be summarised as follows:

— An annotation strategy based on dual bounding boxes to improve the robustness of gate detection under perspective variations.

— We present a geometric reconstruction method that derives a polygonal representation of the gate from dual bounding-box detections.

— To demonstrate the effectiveness of the detections, we conduct experiments in both simulated and real-world environments, showing detection performance and speed during a drone race.

The rest of this paper is organised as follows: Section 2 presents related work in object detection and drone races. Section 3 details the methodology used for polygonal representation from dual bounding box annotations.

The experiments are presented in Section 4, which shows qualitative and quantitative detection results. Conclusions and future work are presented in Section 5.

2 Related Work

In recent years, autonomous drone racing has emerged as a research field involving optimal control, localisation, autonomous navigation, and object detection. The latter plays a key role in UAV-based gate detection, enabling the identification using convolutional neural networks (CNNs) as detectors under conditions that often remain challenging for classical vision-based approaches. Detection architectures such as Single Shot Detector (SSD) [8], R-CNN [32], and PencilNet [26] have demonstrated suitable results in controlled scenarios.

Other works have addressed object representation beyond conventional bounding boxes. Corner2Net [18] formulates object detection as a corner localisation problem, estimating object boundaries through cascaded corner predictions. In [11], the author reconstructs polygonal shapes exploiting geometric relationships among detected line segments, whereas CenterPoly [24] performs instance segmentation, predicting object polygons rather than bounding boxes. Similarly, PolyGeom [23] employs a geometry-aware graph transformer to extract polygons from remote sensing imagery, modelling spatial and structural relationships.

Recent YOLO-based variants have also explored alternative object representations, such as YOLO-Pose [19], which extends detection to keypoint estimation, enabling the localisation of structured landmarks. YOLO-OBB [10] incorporates oriented bounding boxes to represent rotated objects better, whereas YOLO-Segmentation methods [15] generate pixel-wise object masks for instance segmentation tasks. Although these approaches can provide more geometric information than conventional detectors, they require specialised annotations, additional computational resources, and more

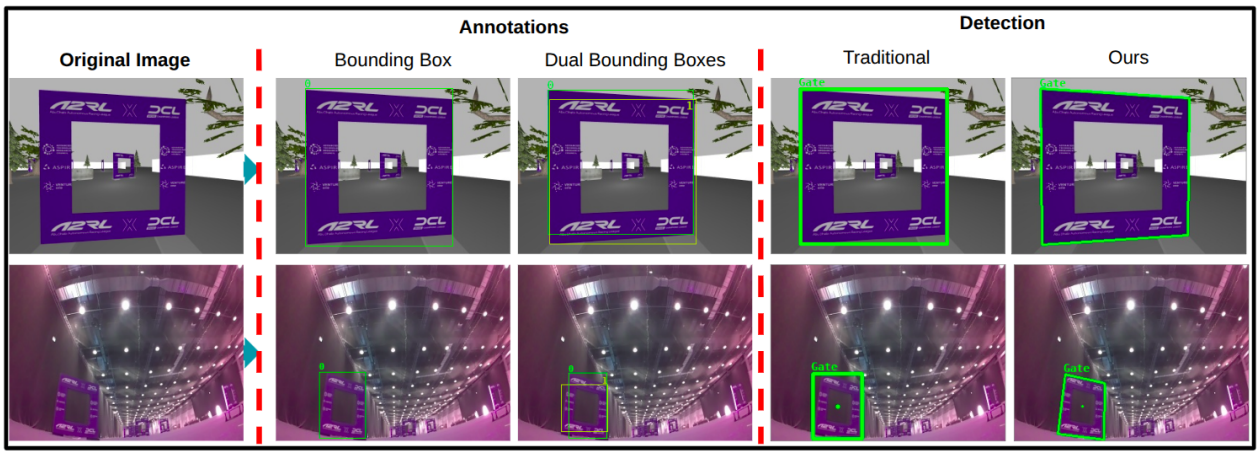

Fig. 1. Conventional methods use bounding boxes for detection, but they fail to represent the gate's geometric shape from different perspectives. Our method employs dual bounding-box detections to reconstruct an accurate representation of the gate, leveraging corner information for precise detection

complex training procedures, limiting their applicability in real-time robotic systems.

Despite these advances, approaches still rely on conventional bounding-box detection, in which the gate is represented by a single rectangular region regardless of its orientation or perspective. This limitation can reduce accuracy, particularly when the geometric form is not properly represented. In the first steps, [3] employed an event-based method to improve detection in different perspectives during gate localisation. However, the method relies on event cameras and additional perception algorithms, increasing computational requirements. The author in [14] provides an overview of autonomous drone racing in competitions such as IROS and AlphaPilot, highlighting the strategies explored to address the challenges of high-speed autonomous flight.

On the one hand, approaches such as GateNet [25] and DeepPilot [28] use CNNs to perceive gates from a single monocular image. The first method divides the image into rows and columns to estimate pose offsets, thereby improving the geometric form and perspective detection. DeepPilot processes a mosaic of frames using a distilled PoseNet-based architecture to generate flight commands for autonomous navigation, while using an SSD architecture for detection.

Nevertheless, the performance fails when the gate exhibits skewed perspectives.

On the other hand, [17] divided the image into multiple sections to filter incomplete predictions and reconstruct missing polygons. Similarly, [12] defined vector fields along the edges of the gate to connect the corners of the structure by Part Affinity Fields. Meanwhile, works such as [13, 16] apply image-mapping and distortion policies around the gate region to improve robustness to disturbances and visual distractors that may affect detection.

In contrast, [31, 33] propose techniques based on waypoint navigation flights for gate detection across different perspectives and orientations. Other works, [27, 30], apply continual learning and incorporate 3D occupancy representations as a voxel grid to adapt detection to diverse illuminations, distances, and orientations, improving detection results. Similarly, state estimation approaches have been employed to optimise flight trajectories through racing circuits [29].

Also, geometric consistency is essential for achieving accurate and reliable autonomy in race scenarios. This has motivated the development of perception pipelines based on stereo vision, where gates are detected from grayscale images.

In [7], the image from the right camera is used as it is closer to the centre of the gate, improving

alignment during detection. Likewise, [9] estimates the structure of gates by interpolating predicted poses and orientations, enabling a geometric representation during autonomous flight.

Despite all of the above, A2RL $\times$ DCL has driven the development of new perception and control strategies to integrate artificial intelligence into aggressive flight. Thus, [5] employs YOLOv8-Pose for gate detection, predicting bounding boxes around the inner corners for fast, efficient perception. Likewise, [22] uses a landmark-based detector to estimate UAV poses and orientation using gates with known spatial locations, combining detections with visual-inertial odometry.

In [6] uses GateNet with QuAdGate to extract edges from segmentation masks and compute the gate corners, applying Homography-based outlier rejection to improve the geometrical consistency.

Finally, [4] associates gate detections with mapped representations, fusing spatial consistency and Euclidean distances between detections and projected centroids.

Although these approaches achieve accurate and robust gate perception across varying perspectives, they rely on additional perception modules or sensor fusion pipelines. In contrast, a lightweight, efficient, perspective-aware detector could simplify the perception process while retaining robustness in real-time. For this reason, the present work explores an alternative based on YOLO to reconstruct a polygonal gate representation from dual bounding-box detections.

3 Methodology

This work aims to develop an accurate and efficient gate detector that is robust to perspective changes, orientation variations, and scale differences. The proposed methodology involves developing an annotation strategy and a polygonal reconstruction based on the geometric structure of gates. Thus, the methodology consists of generating dual bounding-box annotations to train a YOLO network.

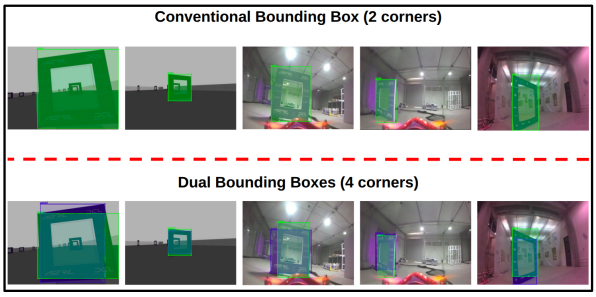

Fig. 2. Comparison between single and dual bounding boxes, where our strategy identifies the four corners

3.1 Dual Bounding Box Strategy

The dataset used in this work was collected with a monocular camera onboard the UAV at a resolution of 640×480 and a frequency of 30 Hz across three environments. At the same time, a human pilot manually controlled the UAV, capturing multiple gate perspectives, variations in scale, rotation, and view angles, as well as oblique configurations.

We used the Robot Operating System (ROS) to establish communication between the ground control station (GCS) and the onboard computer and acquired multiple ROSbag sequences during the flights. In this way, we collected 5 K training images, 500 for validation, and 500 for testing for both environments 1 and 2, while we collected 12 K training images, 1200 for validation, and 1200 for testing in environment 3.

Furthermore, we observed that the gate structure maintains a square geometry regardless of perspective, but changing the gate's appearance into irregular shapes. These have both small and large side structures due to perspective distortion in the image. In this way, we assume that the gate geometry can be reconstructed from the four corners, which is difficult because conventional bounding boxes define only two corners: $(xmin, ymin)$ and $(xmax, ymax)$. However, we can manually annotate both sides of each gate with two bounding boxes to represent the geometry using the LabelImg tool.

Therefore, we introduce a dual bounding-box annotation strategy to represent the four corners, detecting its left and right edges regardless of perspective. We manually annotated each

gate with two classes: gate1 and gate2, where gate1 contains to the $(x1, y1, x3, y3)$ and gate2 $(x2, y2, x4, y4)$ corners. Figure 2 illustrates the difference between a single and dual bounding box under different perspectives, where traditional annotations enclose the entire gate within a single region. In contrast, our strategy separates the gate's sides into independent bounding boxes, enabling a more accurate representation of its geometry.

In this way, we created a dataset with multiple perspectives, orientations, and scale variations, preserving the structural information needed to estimate the four corners. Thus, these annotations provide sufficient geometric information to reconstruct the gate shape from detections without relying on additional methods. Once the detector predicts both bounding boxes, the four corners of the gate can be merged to generate a polygonal representation consistent with the observed perspective. Finally, to maintain detection of dual bounding boxes, it is necessary to visualise the 4 sides of the gate in each image to ensure detection of the gate1 and gate2 classes.

3.2 Geometric Gate Reconstruction

For the training process, we used PyTorch version 2.3 on a workstation running Ubuntu 20.04 equipped with an NVIDIA RTX 4070 GPU and 16 GB of RAM. Besides, we used 100 epochs, a batch size of 16, an imgsz of 640, an SGD optimiser, and a learning rate of 0.01 to train the network. To avoid false positives or present detection with multiple gates, we create a third class corresponding to *background*. This class consists of images of the environment with no gates, multiple gates, three-sided gates, or features that affect detection.

Therefore, we discard the third class during estimation to keep only the gate bounding boxes, making the detection more robust. We decided to incorporate a third class because YOLO detects certain elements in the scenario, leading to false detections. This issue was observed in the first steps, so we decided to create a class that includes elements we don't want in the final detection. In this way, we ensure two detections corresponding to the gate1 and gate2 classes, discarding the background when reconstructing the gate's geometry.

For geometric gate reconstruction, we use the YOLO model to predict dual bounding boxes for each input image, yielding estimates with ID per class, confidence, and corners represented as $xmin, ymin, xmax, ymax$. Additionally, we set a confidence threshold of 0.7 to retain only detections above this threshold, thereby preserving the geometric information for reconstruction.

Thus, we define an ordered coordinate vector composed of the corners of dual bounding boxes to simplify the geometric reconstruction process:

$$P = [x1, y1, x2, y2, x3, y3, x4, y4].$$

Finally, we used the vector P, with 8 coordinates, to reconstruct the gate geometry using OpenCV's polylines function. Thus, we connect the corners to form a polygon, calculate a centroid, and draw the gate's accurate shape regardless of perspective. Furthermore, we have real-world measurements of the gate and use them to compute the inner polygon that the UAV must pass through. Figure 3 shows the reconstruction of both the outer and inner borders of the gate, providing an accurate representation of the space through which the UAV can safely navigate without colliding with it.

4 Experiments and Results

We conducted two experiments to evaluate the effectiveness of the strategy under three different environments: 1) A simulated scenario with 23 gates in the Gazebo simulator, 2) an indoor/outdoor environment with 2-4 racing gates similar to the competition, and 3) the official racing circuit with 11 gates (Figure 4).

In the first experiment, we measure gate detection performance using five metrics on the test dataset for each environment and show visual results of geometric reconstruction. We also compared our strategy against a conventional bounding box and a YOLO-Seg model, measuring the mean IoU of the polygon reconstruction. Finally, we evaluate processing speed in frames per second (fps) using YOLOv8 and YOLOv11 models.

ISSN 2007-9737

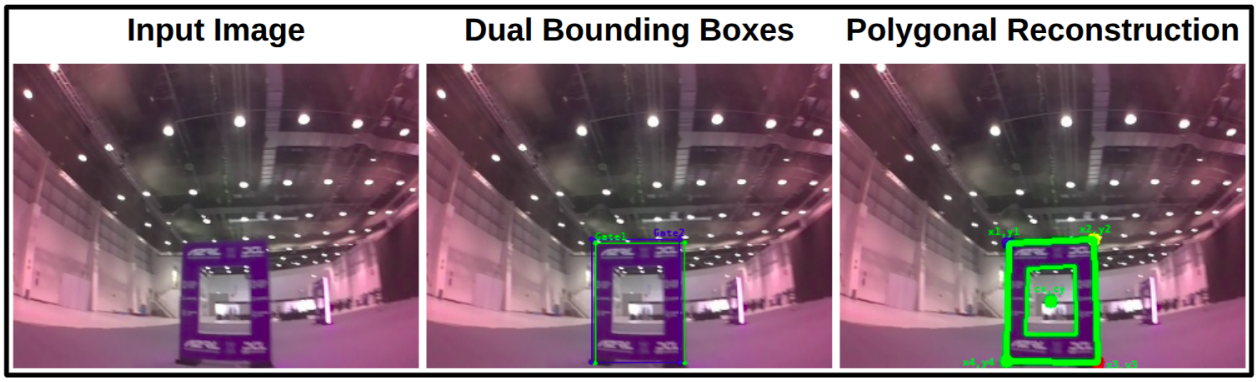

Fig. 3. Geometric gate reconstruction from the four corners obtained through dual bounding-box detection

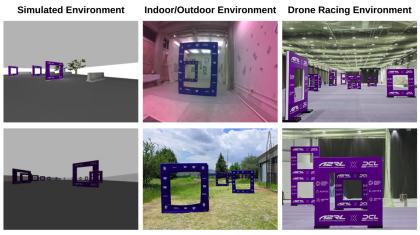

Fig. 4. Environments used to evaluate gate detection using a single and dual bounding boxes

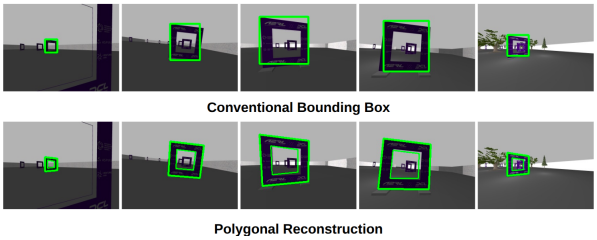

Fig. 5. Detection results on a simulated environment using a single bounding box and our strategy for geometric reconstruction based on dual bounding boxes

4.1 Gate Detection Performance Results

We evaluate our strategy across three scenarios to demonstrate its effectiveness and geometric gate reconstruction, using five detection metrics and visual results compared to conventional bounding box detection. For a fair comparison, the environments present the same configuration under varying lighting conditions, perspectives, and scales. This means that both models were evaluated in the same situations in which the gate appeared distant, rotated, or irregularly formed. Thus, we use the following metrics: Precision, Recall, F1-Score, mAP@0.5, and mAP@0.5:0.95, where TP denotes correct detections, FP denotes false detections, and FN denotes instances where there was a gate, but the model didn't detect it.

These results are presented in the Table 1.

In Figure 5, we present detection results with a single bounding box and the geometric reconstruction using a dual bounding box in the simulated environment. The qualitative results include different perspectives, such as rotations, distance scales, and inclinations, in which our strategy accurately captures the gate's form. We can visualise that a conventional detector identifies the gate and encloses it in a rectangular region without representing its geometric shape.

This shows a suitable detection but does not represent the gate's correct geometric form, which our strategy interprets well.

For the second scenario, we evaluate the approaches in real indoor and outdoor settings to assess the robustness under varying lighting conditions. This scenario exhibits abrupt changes in illumination: indoors, it appears dark, while outdoors, sunlight can impair detection. However, both approaches detect the gates correctly, while the conventional approach finds a rectangular region for the gate, our strategy reconstructs the geometry from diverse perspectives and inclinations.

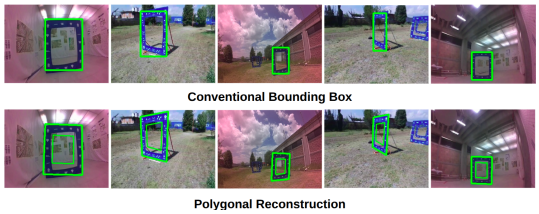

Fig. 6. Detection results on an indoor/outdoor environment using a single bounding box and our strategy based on dual bounding boxes

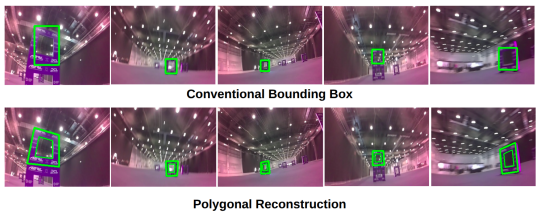

Fig. 7. Detection results on the A2RL $\times$ DCL circuit using a single bounding box and our strategy for geometric reconstruction based on dual bounding boxes

We show the qualitative results for this scenario in Figure 6 and the quantitative results in Table 1, using our strategy and a conventional bounding-box approach for gate detection. Finally, we evaluated our strategy in the third environment, which consists of the drone-race circuit at the A2RL $\times$ DCL site with 11 gates. Nonetheless, the competition indicates that the UAV must complete two laps with a total of 22 gates to detect.

The circuit length exceeds 150 metres, posing a real challenge due to different perspectives of the gate, multiple scales, multiple gates on the track, and varying illumination. Thus, we present qualitative detection results in Figure 7, evaluating detection and geometric reconstruction, in which our strategy maintains an accurate gate shape. In contrast, single bounding-box capture just the region of the gate without incorporating the correct geometric form. In addition, we present the quantitative detection results in Table 1.

4.1.1 Discussions

The results across the three scenarios reported in Table 1 demonstrate the effectiveness of both conventional detection and our proposed strategy. Although the evaluated approaches achieved Precision, Recall, and F1-scores above 0.9, only our strategy reconstructs an accurate gate geometry. This leverage is essential in drone racing applications, where gate detection alone is insufficient, as the UAV must estimate its orientation, perspective, and spatial configuration to generate safe flight trajectories.

Furthermore, this challenge becomes more critical during high-speed manoeuvres, where accurate geometric information is required to avoid collisions and maintain robust navigation. Some approaches, such as Yolo-Obb [10], Yolo-Pose [19], and Yolo-Seg [15], can provide geometric representations. However, they require specialised annotation and additional computational resources. In contrast, the proposed strategy uses bounding-box annotations and remains suitable for deployment on board computers used in drone racing platforms.

To further investigate alternative geometric representations, we evaluated a YOLO-Seg-based approach that segments the entire gate without detecting its corners. Nonetheless, segmentation requires a more demanding annotation process because we need to label the complete object contour. For this experiment, we have annotated 1,700 training images, 150 validation images, and 150 test images and used the same settings employed for the conventional and dual bounding box to train the network. Yolo-Seg produces pixel-wise masks rather than geometric information; we therefore need an additional post-processing step to extract the gate boundaries and estimate the four corners for polygon reconstruction.

In this way, we use the mask pixels generated by Yolo-Seg to locate the four corners of the gate and save their coordinates for geometry reconstruction. In addition, we evaluated a single bounding box, our dual bounding box strategy, and Yolo-Seg for reconstructing the gate's geometry. For evaluation, we compare polygon reconstruction against ground-truth annotations using the IoU metric and report both the mean and standard deviation. To this, we use the following Equation 1 where P_{GT} is the ground-truth polygon and P_{Pred} is the polygon estimated.

Table 1. Gate detection performance across the three test environments using five detection metrics

Environment	Approach	Precision	Recall	F1-Score	mAP@0.5	mAP@0.5:0.95
Scenario 1	One B-box	0.966	0.959	0.962	0.988	0.942
	Dual B-box	0.974	0.974	0.974	0.993	0.964
Scenario 2	One B-box	0.932	0.981	0.956	0.977	0.911
	Dual B-box	0.957	0.964	0.960	0.989	0.956
Scenario 3	One B-box	0.857	0.959	0.905	0.927	0.817
	Dual B-box	0.955	0.960	0.957	0.987	0.953

Table 2. Quantitative evaluation of gate geometry reconstruction using mean IoU and standard deviation

Environment	Approach	Polygon IoU.
Scenario 1	Yolo-Seg	0.921 ± 0.024
	One B-box	0.868 ± 0.015
	Dual B-box	0.973 ± 0.007
Scenario 2	Yolo-Seg	0.314 ± 0.336
	One B-box	0.912 ± 0.025
	Dual B-box	0.947 ± 0.020
Scenario 3	Yolo-Seg	-
	One B-box	0.817 ± 0.101
	Dual B-box	0.955 ± 0.016

Table 3. Average performance speed of the system using two YOLO versions for one and dual bounding boxes detection and polygonal reconstruction

Process	YOLOv8	YOLOv11
One B-box	151 fps	169 fps
Dual B-boxes	143 fps	156 fps
Polygonal Reconstr.	166 fps	179 fps

IoU measures the overlap between the two polygons and computes the ratio of their intersection area to their union area. Higher IoU values indicate a closer agreement between the reconstructed gate geometry and the ground-truth annotation:

$$IoU = \frac{\text{Area}\left(P_{GT} \cap P_{Pred}\right)}{\text{Area}\left(P_{GT} \cup P_{Pred}\right)}. \quad (1)$$

Finally, we present quantitative and qualitative results comparing the gate geometry reconstructions in Table 2, while Figure 8 illustrates representative reconstruction examples across the evaluated environments. We can observe a good representation of the gate's geometry using Yolo-Seg and our strategy in the first two scenarios.

However, the Yolo-Seg model fails to detect and segment the corners in environment 3, indicating that the approach has difficulty capturing the gate's geometry in that scenario.

Finally, we present the segmentation model metrics: Precision 0.967, Recall

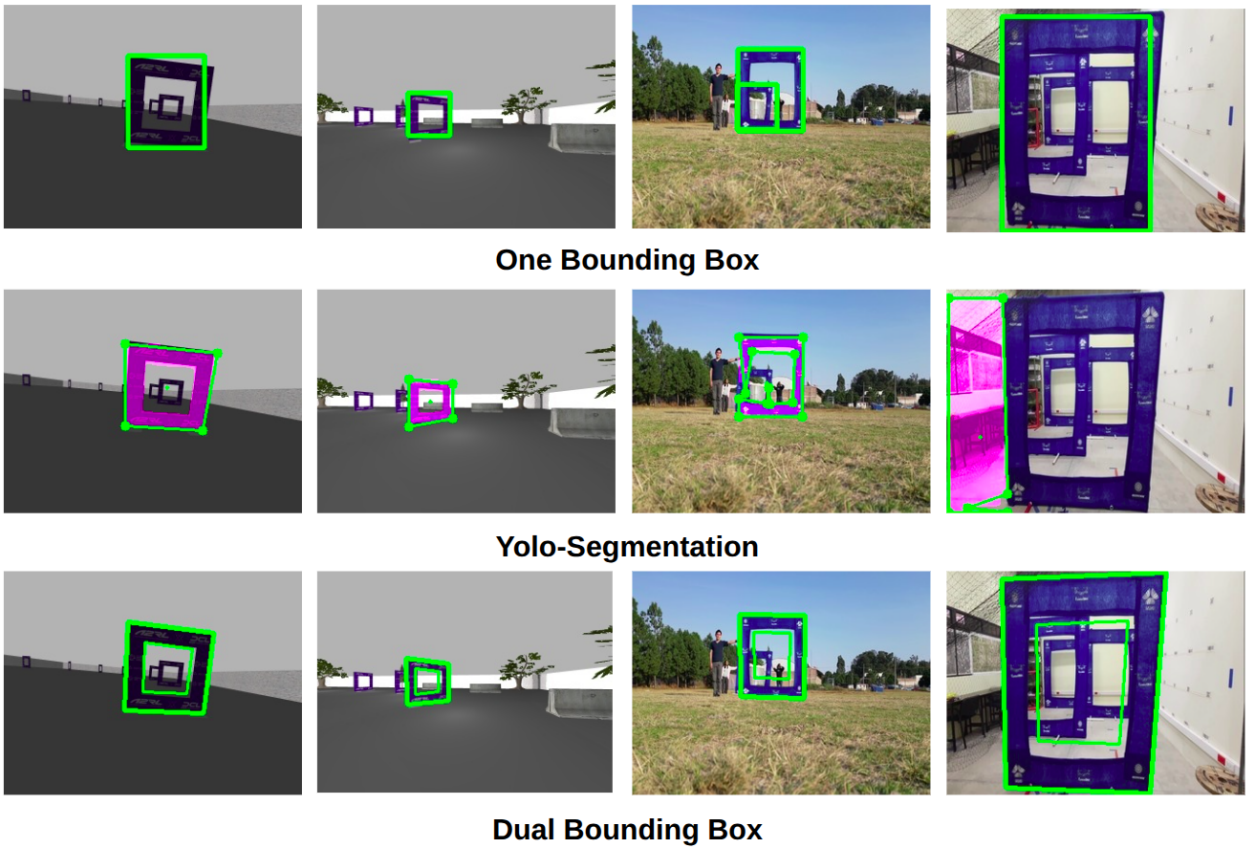

Fig. 8. Polygon reconstruction obtained from One B-box, Dual B-box, and Yolo-Segmentation approaches

0.884, F1-score 0.923, mAP@0.5 0.927, and mAP@0.5:0.95 0.847.

4.2 Performance Results

The second experiment evaluates the system's performance by measuring processing speed during gate detection and polygonal reconstruction. Thus, we conducted two evaluations on the same workstation used for training, comparing the performance of the YOLOv8 and YOLOv11 models using ROSbag sequences collected during flight tests on the racing circuit. These tests consist of a drone race, which allows us to estimate the average processing speed in fps. Table 3 presents performance results for both Yolo versions, demonstrating the feasibility of the proposed method for real-time detection in drone racing applications.

For a fair comparison, we trained both Yolo models using the same dataset, computer, annotation strategy and training parameters. The results show that the YOLOv11-based detector achieved faster inference than YOLOv8 with dual bounding-box detection. Besides, the polygonal process had low computational overhead, preserving real-time performance while reconstructing the gate's geometric form.

These results show that our strategy provides accurate and stable gate detection without requiring additional processing or methods to recover the gate shape as yolo-seg.

In contrast, the conventional detector uses a single bounding box, resulting in slightly faster inference because it predicts only one class.

However, the performance difference was similar to that of our strategy using the dual-bounding-boxes approach. Therefore, a few frames per second represent a worthwhile trade-off in exchange for obtaining a more precise representation of the gate geometry. In addition, our strategy provides improved detection and better geometric interpretability under changes in perspective and orientation, thereby enabling reconstruction of the gate's structure during drone races.

5 Conclusion and Future Work

In this work, we proposed a gate-detection strategy for drone racing that uses dual bounding-box annotations to reconstruct polygonal shapes. The method uses two classes to recover the four corners of the gate and reconstruct its geometric shape without relying on additional modules. Besides, we used the corners to generate an inner polygonal representation of the gate, enabling the UAV to identify the traversable region for flying. The main contribution of this work is a dual bounding-box strategy that improves robustness to variations in perspective, orientation, and scale.

To ensure the effectiveness of our approach, we presented both quantitative and qualitative experiments across different scenarios. Additionally, we evaluate the mean IoU by comparing our strategy against a single B-box and YOLO-segmentation for reconstructing the gates' polygonal shapes. In this way, our strategy achieves a mean IoU score of 0.95 across three environments and an average speed performance of 150 fps. This result indicates that the dual-bounding-boxes strategy is suitable and accurate for geometric reconstruction in drone-racing applications.

In addition, our strategy can be extended to other geometric structures, where reconstructing an object from multiple bounding boxes may improve both interpretability and geometric accuracy. Nevertheless, the method focuses on objects with square or rectangular characteristics, as irregular shapes cannot be accurately represented by the proposed strategy. For objects with complex or non-uniform geometries, dedicated polygon-based approaches are more appropriate. Therefore, the main limitation of our method is its applicability only to structures that can be decomposed into two or more bounding boxes for polygon reconstruction.

As future work, we want to extend the strategy to other geometric shapes using multiple bounding boxes to reconstruct polygonal forms. Furthermore, we aim to integrate the method into high-speed drone racing scenarios, where precise detection and real-time performance are necessary for autonomous flight navigation.

As well as using the method on other platforms and onboard computers for fast detection.

Acknowledgments

The authors, Aldrich Cabrera-Ponce, Oyuki Rojas-Perez, and Nilda Xolo-Tlapanco, thank the Secretaría de Ciencia, Humanidades, Tecnología e Inovación (SECIHTI) for their scholarships with numbers 802791, 924254, and 1230561.

References

1. **a2rl (Accessed: 2026-03-22).** A2rl autonomous drone championship. https://a2rl.io/autonomous-drone-race/.
2. **AlphaPilot (Accessed: 2026-03-22).** Alphapilot – lockheed martin ai drone racing innovation challenge. https://www.herox.com/alphapilot/.
3. **Andersen, K. F., Pham, H. X., Ugurlu, H. I., Kayacan, E. (2022).** Event-based navigation for autonomous drone racing with sparse gated recurrent network. 2022 European Control Conference (ECC), IEEE, pp. 1342–1348.
4. **Azhari, M. B., Han, D., Park, S., Shim, D. H. (2026).** Robust tightly-coupled filter-based monocular visual-inertial state estimation and graph-based evaluation for autonomous drone racing. arXiv preprint arXiv:2603.02742.
5. **Azhari, M. B., Han, D., You, J. I., Park, S., Shim, D. H. (2025).** Drift-corrected monocular vio and perception-aware planning for autonomous drone racing. arXiv preprint arXiv:2512.20475.

6. **Bahnam, S. A., Ferede, R., Blaha, T. M., Lang, A. E., Lucassen, E., Missinne, Q., Verraest, A. E., De Wagter, C., de Croon, G. C. (2026).** Monorace: Winning champion-level drone racing with robust monocular ai. arXiv preprint arXiv:2601.15222.

7. **Bosello, M., Pinzarrone, F., Kiade, S., Aguiari, D., Keuter, Y., AlShehhi, A., Caminati, G., Wong, K. L., Chou, K. S., Halepota, J., et al. (2026).** On your own: Pro-level autonomous drone racing in uninstrumented arenas. IEEE Robotics and Automation Letters, Vol. 11, No. 3, pp. 2674–2681.

8. **Cabrera-Ponce, A. A., Rojas-Perez, L. O., Carrasco-Ochoa, J. A., Martinez-Trinidad, J. F., Martinez-Carranza, J. (2019).** Gate detection for micro aerial vehicles using a single shot detector. IEEE Latin America Transactions, Vol. 17, No. 12, pp. 2045–2052.

9. **Carlotti, N., Antonazzi, M., Cereda, E., Nava, M., Basilico, N., Palossi, D., Giusti, A. (2026).** Self-supervised domain adaptation for visual 3d pose estimation of nano-drone racing gates by enforcing geometric consistency. arXiv preprint arXiv:2603.02936.

10. **Dong, J., Deng, L., Wan, D., Liu, C., Yin, J., Guo, M., Zhang, H., Lin, S., Liu, H., Liu, L. (2025).** Oriented bounding box detection algorithm for dense scenarios of robotic arm operation. Expert Systems with Applications, pp. 129678.

11. **Ferreira Jr, A., Fonseca, M. J., Jorge, J. A. (2023).** Polygon detection from a set of lines. arXiv preprint arXiv:2312.16363.

12. **Foehn, P., Brescianini, D., Kaufmann, E., Cieslewski, T., Gehrig, M., Muglikar, M., Scaramuzza, D. (2022).** Alphapilot: Autonomous drone racing. Autonomous Robots, Vol. 46, No. 1, pp. 307–320.

13. **Fu, J., Song, Y., Wu, Y., Yu, F., Scaramuzza, D. (2023).** Learning deep sensorimotor policies for vision-based autonomous drone racing. 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, pp. 5243–5250.

14. **Hanover, D., Loquercio, A., Bauersfeld, L., Romero, A., Penicka, R., Song, Y., Cioffi, G., Kaufmann, E., Scaramuzza, D. (2024).** Autonomous drone racing: A survey. IEEE Transactions on Robotics, Vol. 40, pp. 3044–3067.

15. **Kang, M., Ting, C.-M., Ting, F. F., Phan, R. C.-W. (2024).** Asf-yolo: A novel yolo model with attentional scale sequence fusion for cell instance segmentation. Image and Vision Computing, Vol. 147, pp. 105057.

16. **Kaufmann, E., Bauersfeld, L., Loquercio, A., Müller, M., Koltun, V., Scaramuzza, D. (2023).** Champion-level drone racing using deep reinforcement learning. Nature, Vol. 620, No. 7976, pp. 982–987.

17. **Kazim, M., Zaidi, A., Ali, S., Raza, M. T., Abbas, G., Ullah, N., Al-Ahmadi, A. A. (2022).** Perception action aware-based autonomous drone race in a photorealistic environment. IEEE Access, Vol. 10, pp. 42566–42576.

18. **Liu, C., Liu, J., Wei, H., Yang, J., Xu, L., Guo, Y., Fang, L. (2024).** Corner2net: Detecting objects as cascade corners. arXiv preprint arXiv:2411.15772.

19. **Maji, D., Nagori, S., Mathew, M., Poddar, D. (2022).** Yolo-pose: Enhancing yolo for multi person pose estimation using object keypoint similarity loss. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 2637–2646.

20. **Moon, H., Sun, Y., Baltes, J., Kim, S. J. (2017).** The iros 2016 competitions [competitions]. IEEE Robotics & Automation Magazine, Vol. 24, No. 1, pp. 20–29.

21. **NeurIPS (Accessed: 2026-03-22).** Game of drones: A NeurIPS 2019 competition. https://microsoft.github.io/AirSim-NeurIPS2019-Drone-Racing/.

22. **Novák, F., Petrlík, M., Novosad, M., Gupta, P. M., Pěnička, R., Saska, M. (2026).**

Vision-only uav state estimation for fast flights without external localization systems: A2rl drone racing finalist approach. arXiv preprint arXiv:2602.01860.

23. **Pei, H., Li, H., Hou, X., Yang, B., Deng, H. (2026).** Polygeom: Geometry-aware graph transformer for building polygon extraction in remote sensing images. Remote Sensing, Vol. 18, No. 4, pp. 551.

24. **Perreault, H., Bilodeau, G.-A., Saunier, N., Héritier, M. (2021).** Centerpoly: Real-time instance segmentation using bounding polygons. Proceedings of the ieee/cvf international conference on computer vision, pp. 2982–2991.

25. **Pham, H. X., Bozcan, I., Sarabakha, A., Haddadin, S., Kayacan, E. (2021).** Gatenet: An efficient deep neural network architecture for gate perception using fish-eye camera in autonomous drone racing. 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, pp. 4176–4183.

26. **Pham, H. X., Sarabakha, A., Odnoshyvkin, M., Kayacan, E. (2022).** Pencilnet: Zero-shot sim-to-real transfer learning for robust gate perception in autonomous drone racing. IEEE Robotics and Automation Letters, Vol. 7, No. 4, pp. 11847–11854.

27. **Qiao, Z., Pham, X. H., Ramasamy, S., Jiang, X., Kayacan, E., Sarabakha, A. (2024).** Continual learning for robust gate detection under dynamic lighting in autonomous drone racing. 2024 International Joint Conference on Neural Networks (IJCNN), IEEE, pp. 1–8.

28. **Rojas-Perez, L. O., Martinez-Carranza, J. (2022).** Leveraging a neural pilot via automatic gain tuning using gate detection for autonomous drone racing. 13th International Micro Air Vehicle Conference at Delft, the Netherlands, pp. 110–118.

29. **Römer, R., Emmert, T., Schoellig, A. P. (2025).** Flying through moving gates without full state estimation. 2025 IEEE International Conference on Robotics and Automation (ICRA), IEEE, pp. 9906–9912.

30. **Srivastava, K., Kulkarni, R., Velmurugan, M., Sanker, N. J. (2025).** Vizflyt: Perception-centric pedagogical framework for autonomous aerial robots. 2025 IEEE International Conference on Robotics and Automation (ICRA), IEEE, pp. 9920–9927.

31. **Verraest, A., Bahnam, S., Ferede, R., de Croon, G., De Wagter, C. (2025).** Skydreamer: Interpretable end-to-end vision-based drone racing with model-based reinforcement learning. arXiv preprint arXiv:2510.14783.

32. **Zhang, D., Doyle, D. D. (2020).** Gate detection using deep learning. 2020 IEEE Aerospace Conference, IEEE, pp. 1–11.

33. **Zhao, F., Mei, J., Zhou, J., Chen, Y., Chen, J., Li, S. (2025).** Gate-aware online planning for two-player autonomous drone racing. 2025 IEEE International Conference on Robotics and Automation (ICRA), IEEE, pp. 8536–8542.

Article received on 07/03/2026; accepted on 18/07/2026.
** Corresponding author is Jose Martinez-Carranza.*