Real-time Object Detection in MAVs For Primitive Single-Colored Objects through Intelligent Image Processing combining YOLOV8 and Old-fashioned CV techniques

Hadi Firouzi
Department of Electrical Engineering
Iran University of Science and Technology
Tehran, Iran
hadi.frz.2001@gmail.com

Alireza Abbasi
Department of Electrical Engineering
Iran University of Science and Technology
Tehran, Iran
a_abbassi@elec.iust.ac.ir

Mobina Lashgari
Department of Electrical Engineering
Iran University of Science and Technology
Tehran, Iran
mobina.lashgary@yahoo.com

Alireza Mikaeili
Department of Electrical Engineering
Iran University of Science and Technology
Tehran, Iran
alireza.mikaeili1380@yahoo.com

***Abstract*—In autonomous robotics, a principal challenge is detecting objects in real time. Formerly, classical methods were used for this purpose which are proven not resilient. Varying lighting conditions, complex environments, and instability of MAVs are reasons for this unreliability** [1]**. Yet one of the best practices to detect primitive-shaped single-colored objects is to use old-fashioned CV techniques. In the modern computer vision world, deep learning is increasingly used** [2]**. One of the best deep learning object-detection methods, implemented in different robotic applications is SOTA-YOLO. However it is not optimized for single-colored object detection; here we developed a combinational method of YOLO and an old-fashioned CV technique, implemented in an autonomous MAV. The yielded results prove this method has performed efficiently with a PPV of 0.968 on our gathered Single-Colored-Autonomous-Drone-Race (SCADR) dataset.**

Keywords—MAV, object detection, single-colored object, YOLO, computer vision

I. INTRODUCTION

Partaking in the FIRA-cup 2024 Brazil air autonomous race league was the main motivation of this research, in these competitions, the drone has to take off, navigate through a few single-colored gates, and land on a pre-defined helipad [3]. There was no guiding line between gates, and teams were competing to go through as many gates as possible in the least amount of time with their drones, so a fast and reliable gate detection algorithm was necessary for navigating the drone. Previously most teams relied on old-fashioned computer vision methods to detect the required gates, such as snake-gate which uses the colored pixels to have a prediction on gate-shaped objects and is suitable for on-board processing but has flaws in many areas such as sensitivity to distance from the gate. These methods, however, despite their high-speed performance and low-power hardware requirements [4] are unreliable and have performed weakly in cases of a slight change in the environment's illumination or light reflection on glossy surfaces. Fig. 1 showcases one of these scenarios and old-fashioned ways` lack of ability to detect under these circumstances.

These problems have led us to look for a more reliable, modern method of object detection to be utilized in our drone navigation algorithm. This pursued method of object-detection, if implemented correctly, can out-perform the prior methods of detecting single-colored primitive-shaped objects in not only MAV[1]s but in other image processing applications such as industrial robotic arms [5].

Most of the proposed methods by other researchers, using networks such as ADRnet [6], GATEnet [7], and Pencilnet [8], were developed for the detection of non-single-colored gates and are dependent on the markings on the gate.

Here in this paper, we propose a unique off-board approach to solving the gate-detection problem, which can be relied on to detect under various lighting conditions. The precision of this method should also allow us to calculate the angle of the detected gate relative to the drone itself.

We opted to use YOLOv8 with a combination of prior old-fashioned CV methods, this will allow us not only be able to detect all the gate-like figures in an image but also to be able to confidently eliminate most of the false positives that may be caused by YOLO's limitation. To have a small YOLO model without facing over-fitting, the trained model should've been obtained from a small dataset, respectively not many epochs for training the model should be used. Therefore many other objects might be detected as gates, especially gate-shaped objects like widows or building frames.

[1] Micro Air Vehicle

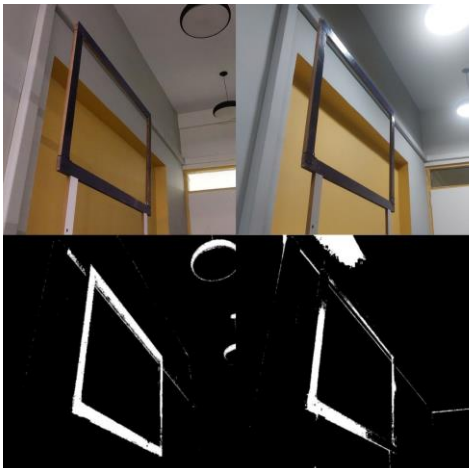

Fig. 1. Old-fashioned CV methods are unreliable in case of a light change in the environment.

II. Explanation of the Proposed Method

Our method consists of a combination of Ultralytics' YOLOv8 [9] and OpenCVs [10] Python libraries.

In this method the output image of the drone, in this case, a Parrot Bebop2 drone (Fig. 2), first passes through the YOLOv8 custom-trained model, then each of the detected objects by the YOLO model is fed in a color filter in HSV color coordinate, knowing that the desired gates are a specific color [3], this will allow us to eliminate almost all the false-positives that might be detected by YOLO model. Fig. 4 showcases the overall process of the proposed method. We will call this proposed method YOLO-MCC by abbreviation from now on.

Fig. 2. Parrot Bebop2

Fig. 3. New proposed method performs better in the same environmental change.

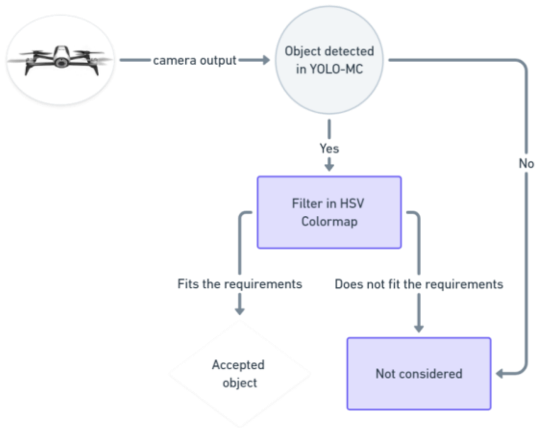

Fig. 4. You-Only-Look-Once Multi-Color-Combination (YOLO-MCC) overall process

In contrast, another approach might be using the YOLO model by itself. This has been experimentally proven to be non-reliable due to amount of the false positives produced by the model in situations where the drone sees a busy and/or noisy environment. To compare the accuracy and efficiency of the proposed method, we will prepare a separate algorithm using this YOLO-only algorithm. Fig. 5 shows the process of this YOLO-only algorithm. This method will be called YOLO-ON in the following.

In these methods, the YOLO models should be trained separately. This is caused by the fact that in the combinational method, we rely on the YOLO to only detect all and any of the possible gates, therefore we will train this model with different gate colors, including gates that are multi-colored, see Fig. 6, on the other hand, the YOLO-ON method will have to be trained on single-colored gates, the same color we will expect it to detect.

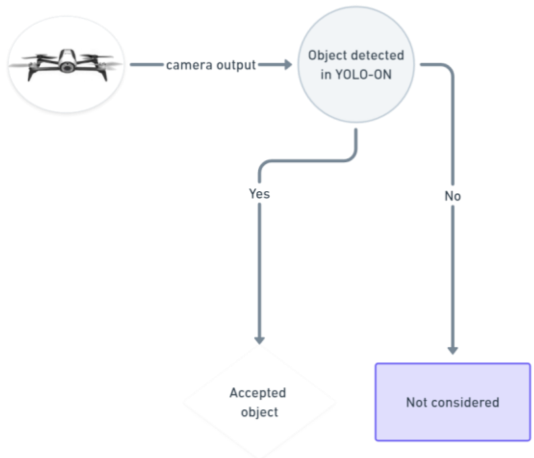

Fig. 5. You-Only-Look-Once Only (YOLO-ON) overall process

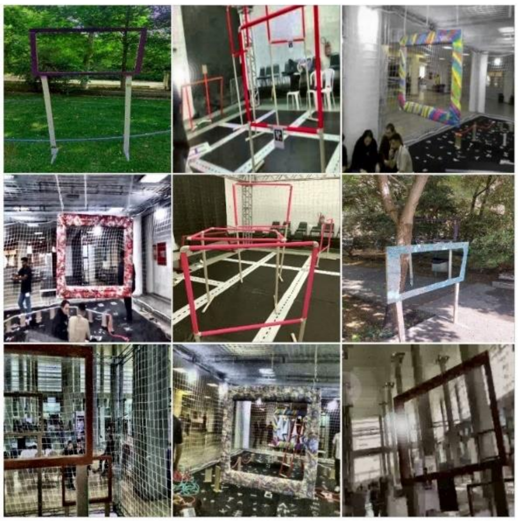

Fig. 6. Some gate examples in the multi-color model's training dataset

To observe the combination's effect on the accuracy of the methods, we will also be implementing a method using only the YOLO model trained for the YOLO-MCC method by itself, this method is expected to produce a massive number of false positives due to the vast multi-colored gate dataset that it will be trained on. This will be called YOLO-MCON.

III. Data Preparation and Model Training

Two sets of gate datasets will be needed, one of which will only include single-colored gates, with a specific color, in this case a dark shade of purple, the second dataset will have multi-colored gates in addition to single-colored ones with different color schemes. Each will be generated using 60 unique annotated images, that will be augmented to 1694 annotated images with a custom augmentation python script.

The augmentation script uses key-point annotations, applies Gaussian blur, hue saturation value shift, and RGB shift, and adds sun flare to annotations randomly. Keep in mind that for YOLO-ON the augmentation should include restrained hue saturation value shifts and RGB shifts, if not the color of

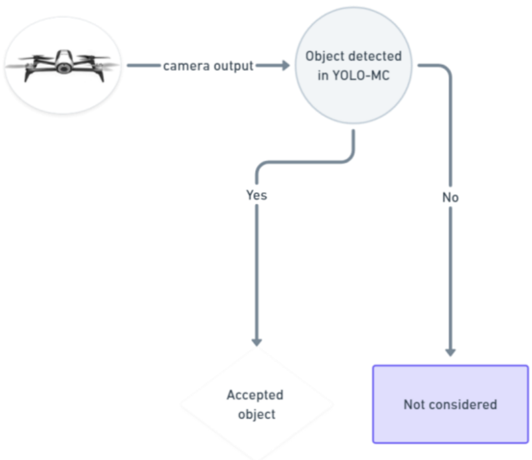

Fig. 7. You-Only-Look-Once Multi-Color-Only (YOLO-MCON) overall process

the gates will be changed which is not desired to be used in this model's training.

In the training process of these two models, we used 100 epochs, which is concluded from our observations in practice.

IV. Method Comparison and Conclusion

After training the models and implementing the algorithms, it's time to compare these methods. We gathered a set of images of the desired gates, detecting these gates was the main goal of this research. 255 new images were taken from the gates and ran through all the methods, this will allow us to compare the precision and operational ability of methods.

The YOLO-MCC method produced the least number of false positives, resulting in the highest Positive Predictive Value (PPV) of 0.968 making it the most desired method for this application. The YOLO-ON method, on the other hand, performed with a PPV of only 0.895. This shows the obvious advantage of the proposed YOLO-MCC method for the application of autonomous drones.

The YOLO-MCON performed with a PPV of 0.797. This method is the least desired method for this application.

Having the numerical results of each method we can conclude that the proposed method of YOLO-MCC, despite producing a significantly higher number of false negatives, had a vivid improvement in PPV, this is highly desirable in applications that are sensitive to corrupt results in object detection such as our application on autonomous MAVs.

Another advantage of the YOLO-MCC method is the easy-to-modify schema of the method, with an easy modification in the color map filter, the same model can be implemented to detect gates that are colored differently from the original gates. In contrast, the YOLO-ON method requires collecting a new dataset of gates, retraining the model, and implementing it, which consumes significant resources and time.

This proposed combination of old-fashioned CV methods with modern State-Of-The-Art object detection algorithms can be implemented in various computer vision applications for the detection of primitive single-colored or specific color-dominated objects. These applications can extend from autonomous robots to industrial automation applications.

V. Future Prospects

The combinational approach of YOLOv8 and traditional CV methods demonstrated promising results for single-colored gate detection in autonomous MAV applications. However, this niche solution has the potential to be expanded in several directions to inspire and support future research:

Real-Time Optimization: The proposed method, though efficient, operates off-board due to computational constraints. Optimizing the pipeline for real-time, on-board processing could further its utility in resource-limited environments.

Applications Beyond MAVs: The robust gate-detection framework could be adapted to various other scenarios, such as industrial robotic arms performing repetitive tasks in environments with changing illumination or underwater drones navigating through obstacles in murky waters.

Generalization to Diverse Geometries: By incorporating advanced data augmentation techniques or transfer learning, this approach could be refined to detect non-primitive shapes or structures with varying geometric complexities.

Public Dataset Contributions: Expanding the SCADR dataset to include varied environmental conditions, object shapes, and sizes could provide a valuable resource for the research community and catalyze advancements in the field.

By exploring these directions, we envision that this combinational methodology can evolve into a more versatile and robust tool, inspiring innovations in the domain of autonomous robotics and beyond.

REFERENCES

[1] A. Mohamed, S. Watkins, R. Clothier, M. Abdulrahim, K. Massey, and R. Sabatini, "Fixed-wing MAV attitude stability in atmospheric turbulence—Part 2: Investigating biologically-inspired sensors," *Prog. Aerosp. Sci.*, vol. 71, pp. 1–13, Nov. 2014, doi: 10.1016/j.paerosci.2014.06.002.

[2] C. Chen *et al.*, "YOLO-Based UAV Technology: A Review of the Research and Its Applications," *Drones*, vol. 7, no. 3, p. 190, Mar. 2023, doi: 10.3390/drones7030190.

[3] S. S. Farzad Ahmadinejad, Mir Alireza Athari, Jacky Baltes, "FIRA Air - Autonomous Race Competition Rules (Pro)," 2019.

[4] S. Li, M. M. O. I. Ozo, C. De Wagter, and G. C. H. E. de Croon, "Autonomous drone race: A computationally efficient vision-based navigation and control strategy," *Rob. Auton. Syst.*, vol. 133, p. 103621, Nov. 2020, doi: 10.1016/j.robot.2020.103621.

[5] M. . Bharath, "Autonomous Landing and Control of a Quadcoptor using Vision Feedback," 2014. [Online]. Available: https://www.researchgate.net/publication/359096227_Autonomous_Landing_and_Control_of_a_Quadcoptor_using_Vision_Feedback

[6] S. Jung, S. Hwang, H. Shin, and D. H. Shim, "Perception, Guidance, and Navigation for Indoor Autonomous Drone Racing Using Deep Learning," *IEEE Robot. Autom. Lett.*, vol. 3, no. 3, pp. 2539–2544, Jul. 2018, doi: 10.1109/LRA.2018.2808368.

[7] H. X. Pham, I. Bozcan, A. Sarabakha, S. Haddadin, and E. Kayacan, "GateNet: An Efficient Deep Neural Network Architecture for Gate Perception Using Fish-Eye Camera in Autonomous Drone Racing," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, Sep. 2021, pp. 4176–4183. doi: 10.1109/IROS51168.2021.9636207.

[8] A. S. Zhongzheng Qiao, Xuan Huy Pham, Savitha Ramasamy, Xudong Jiang, Erdal Kayacan, "Continual Learning for Robust Gate Detection under Dynamic Lighting in Autonomous Drone Racing," 2024.

[9] "YOLOv8 - Ultralytics YOLO Docs." [Online]. Available: https://docs.ultralytics.com/models/yolov8/

[10] "OpenCV-python Documents." [Online]. Available: https://docs.opencv.org/4.x/d6/d00/tutorial_py_root.html