ISSN: 2321-2152

IJMECE

International Journal of modern electronics and communication engineering

E-Mail
editor.ijmece@gmail.com
editor@ijmece.com

www.ijmece.com

ISSN2321-2152
www.ijmece .com
Vol 8, Issue 1, 2020

DETECTION OF OBJECTS USING YOLOV8 DEEP LEARNING

Ms. S. Chandra Priyadharshini, Ms. R. Latha Priyadharshini, Mr. S. Satheesh, Mrs. A. Baranishri
Associate Professor [1], Assistant Professor [2,3,4]
Department of CSE,
chandrapriyadharshini.s@actechnology.in, lathapriyadharshini.r@actechnology.in, ssatheesh@actechnology.in, baranishri.a@actechnology.in

Arjun College of Technology, Thamaraikulam, Coimbatore-Pollachi Highway, Coimbatore, Tamilnadu-642 120

ABSTRACT— The fast development of deep learning in the last ten years has allowed researchers in the area of computer vision to make great strides in the difficult issue of object identification. Researchers The detection accuracy and inference time of an object detector are usually the deciding factors in its assessment. When comparing detection accuracy, two-stage detectors are often better than single-stage detectors, whereas single-stage detectors have shorter inference times. Nevertheless, there have been notable advancements in the field of object identification and related deep learning tasks, such as the YOLO (You Only Look Once) model, which has greatly improved performance. There are two main types of object detectors: those with two stages and those with one stage. When it comes to possible object detection, two-stage detectors utilise intricate structures that zero down on certain regions, whereas single-stage detectors use more straightforward designs that take into account all geographical regions at once. and other architectural offspring, have greatly enhanced detection accuracy to the point that they may sometimes surpass two-stage detectors. The rapid inference times of YOLO models have led to their widespread use, even though these models often sacrifice detection accuracy for speed. Our suggested system incorporates the cutting-edge YOLOv8 model for identification of weapons in real-time. When compared to YOLOv5, YOLOv8 is much faster and more accurate. We have quantized the YOLOv8 model's weights to guarantee efficient performance. Our trials compared the effectiveness of the YOLOv8 and YOLOv5 versions in detecting weapons. By using YOLOv8, we were able to increase our mean Average Precision (mAP) from 89.1% in YOLOv5 to 90.1%. Additionally, we were able to decrease the inference time by 15% when compared to the original YOLOv8 setup by applying weight quantization to the YOLOv8 model. With its enhanced accuracy and accelerated inference, YOLOv8 is a top pick for applications that need real-time weapon identification.

Keywords: Object detection, Single-stage detectors, Two-stage detectors, Deep learning, YOLO, mean Average Precision (mAP).

I.INTRODUCTION

ISSN2321-2152
www.ijmece .com
Vol 8, Issue 1, 2020

When it comes to computer vision, object detection—the process of identifying and accurately localising things inside pictures or videos—presents a daunting barrier. Object detection has evolved into a multipurpose tool with limitless potential due to the proliferation of high-definition cameras and video data. But occlusions, clutter, and the need to recognise several items at once are just a few examples of the intricacies and differences present in real-world settings that make object recognition a challenging operation.

Recent years have seen a dramatic improvement in object identification speed and accuracy because to the introduction of deep learning algorithms. You Only Look Once (YOLO) is one of the most prominent object identification algorithm families; it achieves real-time object recognition by use of deep Convolutional Neural Networks (CNNs). To begin, the input picture is grid-partitioned. Then, for each grid cell, the YOLO algorithm finds the bounding boxes and confidence scores. The unique loss function that YOLO employs is a combination of localization and classification mistakes; this allows it to effectively handle overlapping objects and, in the end, improve detection accuracy. Object detection is a dynamic area where researchers are always working to improve the accuracy and performance of detection systems. Advanced deep learning architectures, more sensor modalities, and merging object identification with other computer vision tasks like instance and semantic segmentation are all possibilities for the future of object detection.

More recently, a new version of the YOLO algorithm called YOLOv8 was released, which substantially increases the speed and accuracy of object recognition. Incorporating characteristics from many levels of abstraction, YOLOv8 unveils a novel architecture based on a hybrid backbone network. By using this novel method, YOLOv8 is able to accomplish accurate object recognition while exhibiting exceptional efficiency. In addition, YOLOv8 has improved training methods, a feature pyramid network, and a revamped anchor box architecture, all of which contribute to its outstanding performance.
With the help of the test set, we can determine how well the suggested method works by computing metrics like accuracy, classification report, and confusion matrix. The model's capacity to categorise user input, custom data points, and random data points is further shown by the accompanying object detection examples.
This study's findings increase object detecting system performance and accuracy, which is a step in the right direction. Through the use of deep learning networks' capabilities.

II.LITERATURE REVIEW

In this work, we presented You Only Look Once, a quick and easy way to identify pictures in real time. The model's primary goals were picture detection (both actual and artistic) and speed. Joe Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi's "You Only Look Once: Unified, Real-Time Object Detection" were the subjects of this literature study.Object identification utilising YOLO based on deep learning. Finally, YOLO brought a single unified architecture for regression, splitting the picture into bounding boxes and computing class probabilities for each. This was in contrast to other object identification algorithms, such as R-CNN. Because of this, YOLO was able to provide more accurate results in a shorter amount of time. Artwork predictions might also be accurate.Joseph Redmon's You Only Look Once: Unified, Real-Time Object Detection. Object detection using a regression technique has been their previous focus. This study proposes the YOLO algorithm as a means to get high accuracy and quality forecasts.

An Analysis of Juan Du's Object Detection Using the CNN Family and YOLO. In this study, the authors provided an overview of object detection families such as CNN and R-CNN, compared their performance, and then presented the YOLO method as a means to improve upon their findings. Matt B. Blaschko's Learning to Localise Objects using Structured Output Regression. Object localization is the subject of this article. In this, they circumvented the limitations of the sliding window approach by localising items using the bounding box method.

III.METHODOLOGY

1. Data preprocessing:

- Gather a labeled dataset containing images or videos with annotated object bounding boxes and class labels.
- Preprocess the data, including resizing images, normalizing pixel values, and augmenting the dataset if necessary to increase diversity.

2. Data Splitting::

- Divide the dataset into three subsets: training, validation, and testing sets. Common splits are 70-80% for training, 10-15% for validation, and 10-15% for testing.

3. Model Selection:

- Choose YOLOv8 as the deep learning model for object detection. You can access pre-trained weights for YOLOv8 or train the model from scratch, depending on your dataset size and task.

4. Model Configuration:

- Configure the YOLOv8 model based on your specific requirements. Adjust parameters such as the number of anchor boxes, network architecture, and input size.

5. Training:

- Train the YOLOv8 model on the training dataset using the selected configuration.
- Utilize a suitable loss function, such as a combination of localization loss and classification loss, which is typically used in YOLO-based models.
- Monitor training progress by evaluating metrics on the validation dataset, such as mean Average Precision (mAP).

6. Model Configuration:

- Fine-tune hyperparameters, including learning rate, batch size, and training epochs, to optimize

ISSN2321-2152
www.ijmece .com
Vol 8, Issue 1, 2020

model performance.

- Implement techniques like learning rate scheduling to aid convergence.

7. Model Evaluation:

- Asses the trained YOLOv8 model on the testing dataset to evaluate its performance on unseen data.
- Custom Data Point Fraud Detection: Prompt the user to input the index of a data point they want to check, extract the corresponding data point from the dataset, make a predicition using the model, and compare it with actual label.

8. Inference:

- Deploy the trained YOLOv8 model for realtime object detection on new images or video streams.
- Measure the inference time to ensure that it meets real-time requirements.

9. Visualization:

- Visualize the object detection results by drawing bounding boxes around detected objects and labeling them with their corresponding class labels.

10. Deployment:

- Deploy the trained YOLOv8 model in your target application, whether it's for surveillance, autonomous vehicles, or any other object detection task.

11. Monitoring and Maintenance:

-

Continuously monitor the model's performance in the production environment and retrain it periodically with new data to maintain accuracy.

12. Documentation:

- Document the entire process, including data collection, model configuration, training parameters, and evaluation results, to facilitate reproducibility and future reference.

IV.IMPLEMENTATION

1. Load the dataset that you want to train your model on. Make sure that the dataset contains images of objects that you want to detect.
2. Annotate the dataset with bounding boxes around the objects you want to detect. You can use tools like labeling, reactlabel for the step.
3. Preprocess the annotated dataset by resizing the images, normalizing the pixel values, and splitting the dataset into training and validation sets.
4. Train the YOLOv8 model on the preprocesses dataset using deep learning framework like pytouch. You can use pre-trained weights to speed up the training process.
5. Evaluate the trained model on the validation set to measure its accuracy and performance. You must use metrics like mean Average Precision (mAP) to evaluate model.
6. Optimize the trained model by fine-tuning its hyperparameters, adjusting the learning rate, or changing the architecture.

ISSN2321-2152
www.ijmece .com
Vol 8, Issue 1, 2020

```
Python    CLI

from ultralytics import YOLO

# Load a model
model = YOLO('yolov8n.pt')  # load an official model
model = YOLO('path/to/best.pt')  # load a custom model

# Validate the model
metrics = model.val()  # no arguments needed, dataset and settings remembered
metrics.box.map    # map50-95
metrics.box.map50  # map50
metrics.box.map75  # map75
metrics.box.maps   # a list contains map50-95 of each category
```

7. Create a new floder and place the data in it and your dataset is ready.
8. Deploy the model on a production environment and use it for object detection tasks.

V.RESULTS

During the initial training phase of the model, the following results were observed.

Fig 1: Trained Data

Fig 2: Load a Model

Fig 3: Comparison with previous YOLO Version

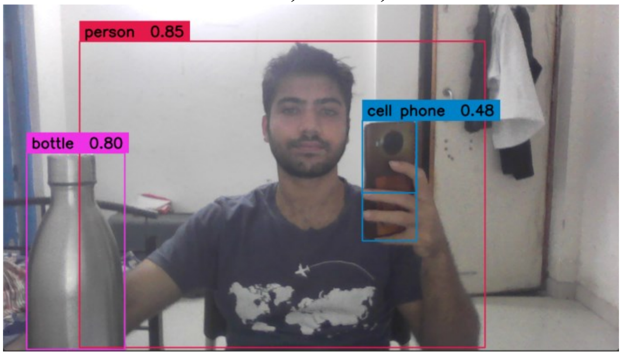
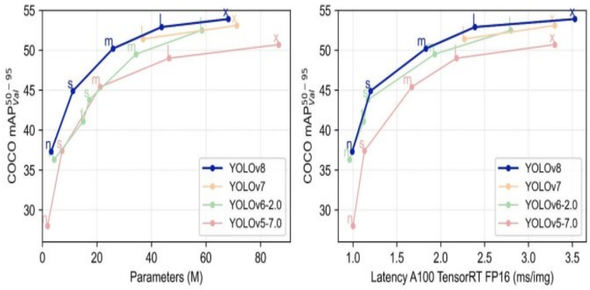


Fig 4: Applications For Object Detection

VI. CONCLUSION

To sum up, YOLOv8's deep learning object recognition is a huge step forward for computer vision applications.



As an algorithm of the YOLO (You Only Look Once) family, YOLOv8 has shown to be very good in efficiently and precisely identifying objects in videos and photos.

Significant advancements in object identification speed and accuracy have been made possible by the use of deep learning techniques like YOLOv8. Thanks to its cutting-edge design and loss algorithms, YOLOv8 can handle challenging real-world situations including occlusions, clutter, and multiple item recognition at once. This makes it an

```
Python    CLI

from ultralytics import YOLO

# Load a model
model = YOLO('yolov8n.yaml')  # build a new model from YAML
model = YOLO('yolov8n.pt')  # load a pretrained model (recommended for training)
model = YOLO('yolov8n.yaml').load('yolov8n.pt')  # build from YAML and transfer weights

# Train the model
results = model.train(data='coco128.yaml', epochs=100, imgsz=640)
```

ISSN2321-2152
www.ijmece .com
Vol 8, Issue 1, 2020

excellent resource for many uses where the ability to identify objects in real-time is critical.

VII. FUTURE ENHANCEMENTS

The release of YOLOv8 marks a major step forward in the ever-increasing development of object identification and deep learning. There are a number of practical uses for it that are about to explode in popularity, such as security, driverless cars, surveillance, and many more. If YOLOv8 and other deep learning-based methods are fine-tuned, they might solve the complicated problems posed by today's dynamic and ever-changing settings, leading to even more progress in object recognition. In addition, by quantifying the weights of YOLOv8, it has been shown that it may decrease inference times while keeping accuracy high.

VIII.REFERENCES

1. M. Y. Zou, J. J. Yu, Y. Lv, B.Lu, W. Z. Chi and L. n. Sun. A Novel Day-to-Night Obstacle Detection Method for Excavators based on Image Enhancement and. IEEE Sensors Journal, 2023, pp. 1–11.

2. C. Li et al., YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications. 2022, [Online]. Available: http://arxiv.org/abs/2209.02976.

3. A. D. Abadi, Y. Gu, I. Goncharenko, and S. Kamijo. Detection of Cyclist's Crossing Intention based on Posture Estimation for Autonomous Driving. 2023, IEEE Sensors Journal, pp. 1–1.

4. J. Redmon and A. Farhadi. YOLOv3: An Incremental Improvement. 2018, arXiv preprint arXiv:1804.02767.

5. S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. 2015, 28, Advances in neural information processing systems.

6. H. Liu, F. Sun, J. Gu, and L. Deng. SF-YOLOv5: A Lightweight Small Object Detection Algorithm Based on Improved Feature Fusion Mode. 2022, Sensors, vol. 22, no. 15, pp. 1–14.

7. Cao Z, Liao T, Song W, Chen Z, Li C (2021) Detecting the shuttlecock for a badminton robot: a YOLO based approach. Expert Syst Appl. https://doi.org/10.1016/j.eswa.2020.113833

8. Albelwi S, Mahmood A (2017) A framework for designing the architectures of deep convolutional neural networks. Entropy 19(6):242.

9. Bhattacharya S, Maddikunta PKR, Pham QV, Gadekallu TR, Chowdhary CL, Alazab M, Piran MJ (2021) Deep learning and medical image processing for coronavirus (COVID-19) pandemic: a survey. Sustain Cities Soc 65:102589. https://doi.org/10.1016/j.scs.2020.102589

10. Quah, Agarwal S, Terrail JO, Jurie F (2018) Recent advances in object detection in the age of deep convolutional neural networks. arXiv preprint arXiv:1809.03193. https://doi.org/10.48550/arXiv.1809.03193