International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 10 Issue: 10 | Oct 2023 www.irjet.net p-ISSN: 2395-0072

Custom Object Detection Using YOLO Integrated with a Segment Anything Model

Yuvaraj Singh S

MCA, Department of Computer Science,
Presidency College, Chennai, Tamil Nadu, India.

Abstract - *Object detection and semantic segmentation are tasks, in computer vision that have applications, such as autonomous driving, surveillance systems and augmented reality. While significant progress has been made in these areas accurately identifying and outlining objects related to a field remains a challenge. In this paper, we propose an integrated approach that combines the efficiency of YOLO (You Only Look Once), an advanced object detection algorithm with the accuracy of the SAM (Segment anything model) Leveraging the strengths of both methods aims to create a system for custom object detection and segmentation that achieves efficient identification and outlining of objects in images, videos and real-time. We provide information about our integrated architecture, including network design, training processes, and inference pipeline. The performance of our approach is evaluated using a custom dataset specifically created for detecting and segmenting field-specific objects. Experimental results show that our integrated system outperforms the YOLO and Segment Anything model (SAM) as well as existing methods in terms of accuracy and efficiency. Our proposed system has potential across domains by addressing the need, for robust and accurate custom object detection and segmentation.*

Key Words: Object Detection, Object Tracking, Segmentation, You Only Look Once (YOLO), Segment Anything Model (SAM).

1. INTRODUCTION

In recent years, computer vision has witnessed significant advancements in the field of object detection and semantic segmentation. Object detection algorithms, such as YOLO (You Only Look Once), have gained popularity for their real-time performance and accuracy. Meanwhile, semantic segmentation models, such as those developed by Meta AI, have showcased impressive capabilities in segmenting objects at the pixel level. In this paper, we propose an innovative approach that integrates the strengths of YOLO and the Segment Anything Model (SAM) from Meta AI to achieve custom object detection and segmentation in images, videos, and real-time.

object detection and segmentation have unique challenges which are to detect and classify objects that may not belong to typical pre-trained classes. Our goal is to develop a system that can detect and classify custom objects accurately while maintaining real-time performance [1]. By combining the high-performance features of YOLO object detection with the pixel-level accuracy of the Segment Anything Model (SAM), we aim to provide a complete solution for object detection and classification which is intended.

The combination of YOLO and the Segment Anything model (SAM) offers several advantages. The YOLO one-step search method enables real-time performance by segmenting objects in a single phase, while the Segment anything model (SAM) excels in capturing fine-grained information for object segmentation accuracy [1]. Combining these methods, we can use the complementary strengths of both methods so that customization and classification processes are more efficient and more accurate.

The contributions of this study include the development of a new architecture that seamlessly integrates the YOLO and SAM models. We analyse training algorithms, architecture, and inference pipelines to ensure efficiency and accuracy. In addition, we test our integrated model with a custom report containing annotated images and videos of various custom products, providing a comprehensive evaluation of its effectiveness and comparing it to the individual YOLO and SAM samples.

2. LITERATURE REVIEW

Custom object detection and segmentation have been areas of active research in computer vision, with several approaches proposed to overcome the challenges of accurately identifying and describing domain-specific features in this section we review the related literature that combines the You Only Look Once (YOLO) Model and Segment Anything Model (SAM). It focuses on identifying and classifying the object.

YOLO has emerged as a popular object detection algorithm due to its real-time performance and high accuracy. The YOLO framework divides an image into a grid and directly predicts bounding boxes and class probabilities from the grid cells [2]. This approach enables efficient object detection without relying on a complex

region proposal network. YOLO versions, such as YOLOv8, have further improved the accuracy and robustness of object detection.

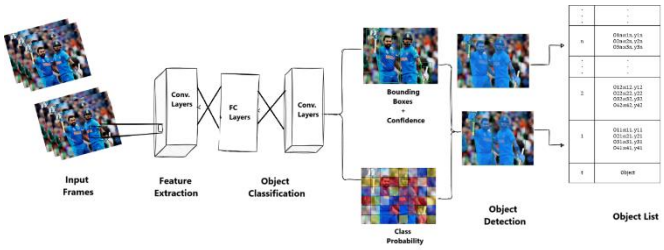

Fig -1: Architectural diagram of YOLOv8.

There has been growing interest in integrating object detection with semantic segmentation to achieve more comprehensive scene understanding in recent years. Semantic segmentation aims to assign pixel-wise labels to an image, providing detailed object masks. One notable method in this regard is the Segment Anything Model (SAM), which utilizes a combination of deep learning techniques to perform accurate and fine-grained segmentation. SAM has been successful in various segmentation tasks, demonstrating its effectiveness in capturing object boundaries and details.

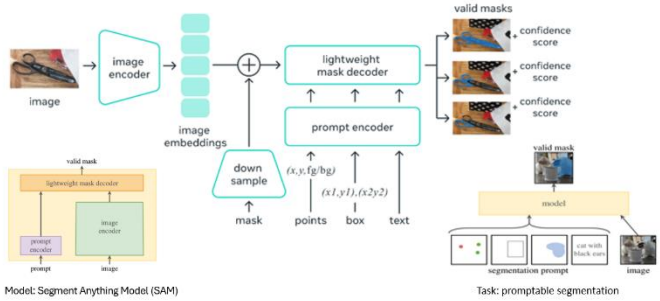

Fig -2: Architectural diagram of SAM.

The integration of YOLO with SAM for custom object detection and segmentation brings together the efficiency of YOLO in object localization and the detailed segmentation capabilities of SAM. By combining these two approaches, the integrated system aims to provide accurate object detection along with precise object masks, specifically tailored to custom object classes.

Several research studies have explored the integration of YOLO with semantic segmentation models SAM. For example, ABC-YOLO introduced an attention-based context-aware module that improves the accuracy of object detection by incorporating semantic information. This integration enhances the detection performance by utilizing contextual cues derived from the segmentation model.

Another notable approach is the Integrated Detection and Segmentation Network (IDSN), which combines YOLO with a deep contour-aware network for simultaneous object detection and instance-level contour segmentation [3]. IDSN achieves state-of-the-art results by leveraging the complementary strengths of object detection and contour-aware segmentation. however, to the best of our knowledge, there is limited prior work specifically focusing on integrating YOLO with the SAM approach for custom object detection and segmentation. This integration provides a unique opportunity to address the challenges associated with accurately identifying and classifying domain-specific features, making the system more accurate, efficient and effective.

In summary, although YOLO has demonstrated its effectiveness and accuracy in object recognition, its combination with the Segment Anything Model (SAM) makes it possible to identify custom objects and advanced segmentation Integrated systems take advantage of the strengths of both methods for accurate detection of object masks. Further studies and experiments are needed to investigate possible sources, to enable detailed analysis and understanding of customized product groups, and to evaluate the performance of this combined approach in custom road object detection and classification services.

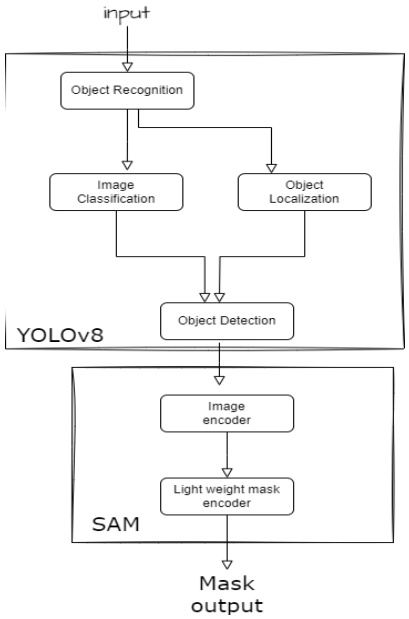

Fig -3: Basic flow diagram of our Integrated Model.

3. METHODOLOGY

Data Collection: Collect a data set containing images or videos of a custom object class of interest. ensure the dataset includes annotations for the object locations and segmentation masks.

Training: To train the YOLOv8 model with a custom dataset of 800 images with subject 2: Rohit and Virat are pre-trained to YOLOv8 which was already pre-trained for the COCO dataset with yolov8x.pt This model was trained for this model 100 epochs via Google Colab [7]. All 800 images were manually annotated using the Roboflow [6]. The data set was trained with the help of the PyTorch library. The images are labelled YOLO. A total of 205 images were used for validation and 595 images were used for training.

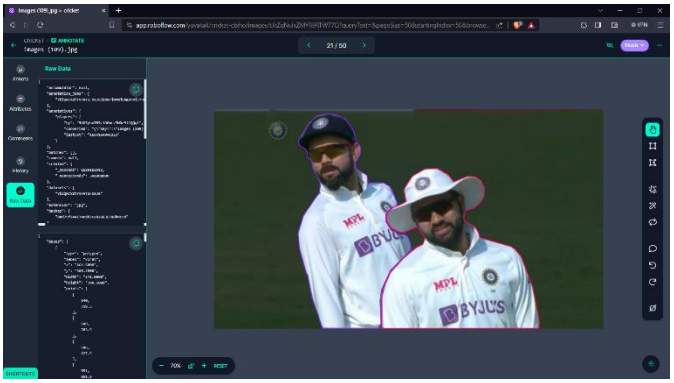

Fig -4: Roboflow tool for Annotation.

To train a model using labeled images, the custom dataset images are in three folders 1. train, 2. test and 3. valid. Each Folder has images with labels. labels are saved in .txt format, The yaml file (custom_data.yaml) specifies the location of the folders to call to train a model.

```
yolo task=detect mode=train model=yolov8x.pt
data= custom_data.yaml epochs=100 imgsz=640
plots=True
```

After the 100 epochs, we get our best model (best.pt). The trained custom dataset using Google Colab (see Fig. 5).

Epochs	Box loss	Class loss	DFL loss (Distributional focal loss)
1	1.618	2.847	1.849
100	0.6364	0.3135	1.084

Table -1: Loss of custom dataset

Segment Anything model (SAM) is already pre-trained with 1+ billion masks and 11 million images of the SA-1B dataset [4]. So that we can use the SAM without labelling them. It segments all the things they've been trained in. Now, we have to segment the custom-specific object so we give our trained model as input and integrate with SAM, so now it only focuses on the model that we trained. So, the latest model is more productive than all other segmentation models.

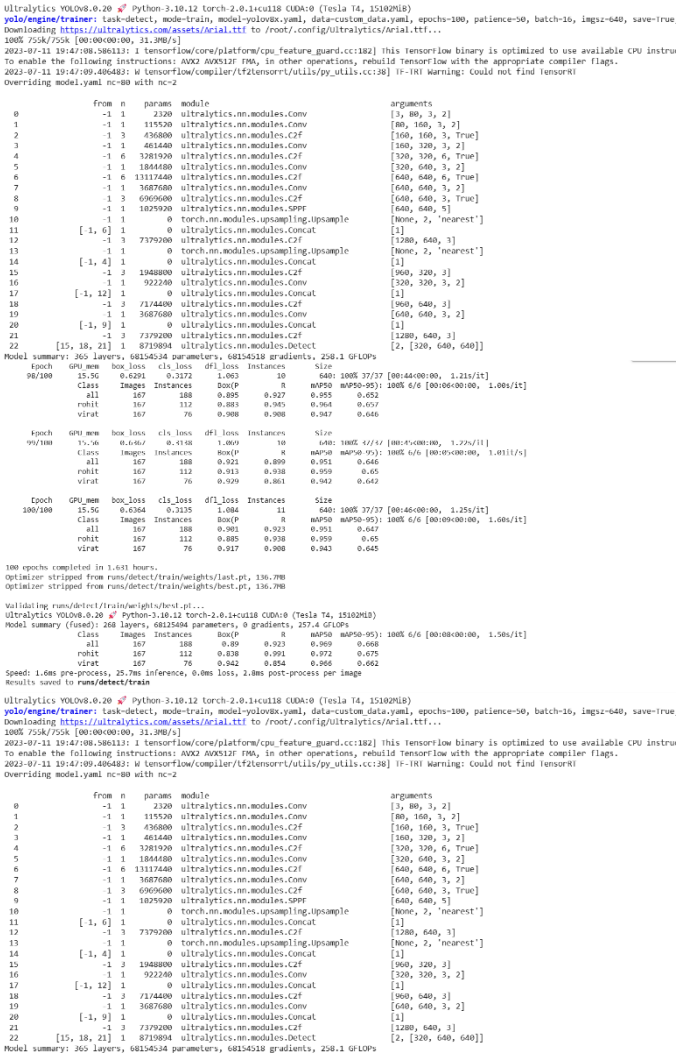

Fig -5: Train the custom dataset using Google Colab.

Fig -6: Trained custom dataset.

4. EXPERIMENTAL RESULTS

Evaluation Metrics: The following metrics are used to evaluate the classification performance of the algorithm:

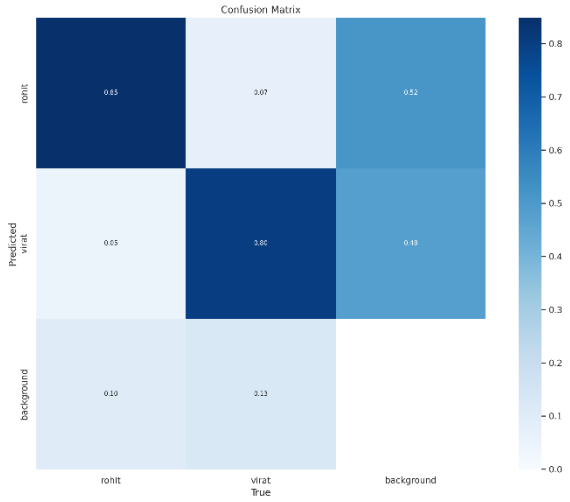

Fig -7: Confusion matrix

4.1 Accuracy: It is defined as the number of correct predictions made by the model over the total number of predictions. This is a good measure, especially when the objective variables in the data are balanced. This can be illustrated as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

where True Positive (TP) is defined as the correct recognition of a training group of objects. A True Negative (TN) is defined as a grossly incorrect unspecified factor, i.e. knowing nothing when something should be known. A false positive (FP) is defined as a false detection, meaning that there is a detection even though no object should be detected. A false negative (FN) is defined as not detecting any ground truth, i.e. the algorithm failed to find the object to be found.

4.2 F1 Score: The precision of the test is determined by the balanced F-measure. If both the false positive and false negative rates are low, the F1 score is considered positive. It is defined as the harmonic medium of precision and recall [5].

$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} = \frac{\text{TP}}{\text{TP} + \frac{1}{2}(\text{FP} + \text{FN})}$$

Where Precision and Recall are defined as follows:

Precision: It is the number of positives divided by the total number of positives predicted by the classifier [5].

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

Recall: It is defined as the number of true positives divided by the sum of true positives and false negatives [5].

$$\text{Recall} = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

Validation: The validation process involves running the model on the validation data set and comparing the model predictions to the ground truth labels. Several metrics such as mean average precision (mAP), intercept over union (IoU), and false positive rate (FPR) can be used to evaluate model performance. Validation results can be used to tune model hyperparameters such as number of trials, batch size, and number of epochs. The goal is to find different hyperparameters that lead to the best performance on the validation dataset.

Fig -8: Validated custom dataset.

We tested a few snapshots of our object detector to check how well it was trained and obtained the precision and recall graph.

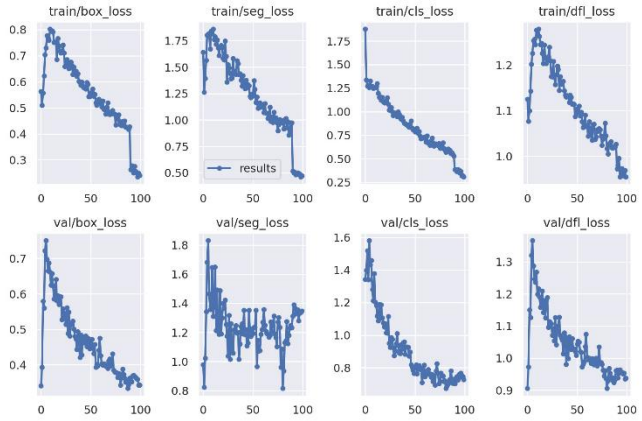

Fig -9: Loss graph for training and validation dataset.

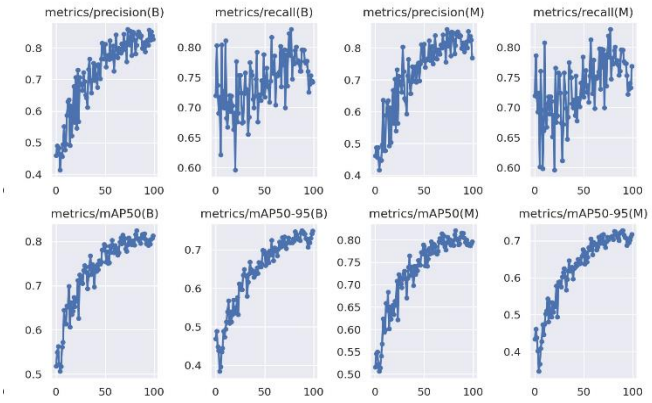

Fig -10: Metrics graph for custom dataset

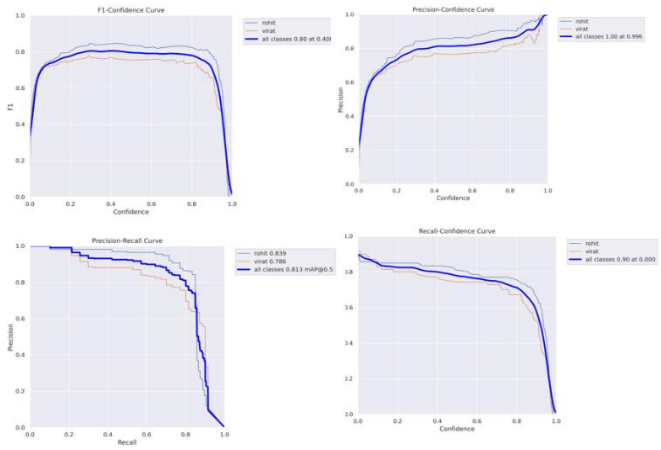

Fig -11: Metrics curve for custom dataset

	P	R	mAP50	mAP50-95
All	0.901	0.923	0.951	0.647
Rohit	0.885	0.938	0.959	0.65
Virat	0.917	0.908	0.943	0.645

Table -2: metrics of custom dataset

When performing visual object detection and segmentation tasks, the video is divided into frames, and each frame and video output is saved with the detection and segment information obtained for each input video after using YOLO and SAM for detection and segmentation. Below are the output screens of the test videos, which provide output as bounding boxes with class names and confidence scores, as well as class segmentation performed by SAM.

Fig -12: Qualitative analysis of the system from video

Normal image SAM image

Yolo + SAM image

Fig -13: Qualitative analysis of the system from images

5. CONCLUSIONS

In this paper, object detection and classification are done by training the detector on images and videos for a custom dataset of 800 images for 2 specific classes Moving object

detection is done by the YOLO detector and the SAM model helps to classify objects to frame a series of different types. While training a detector, accuracy and precision can be achieved by training the system for several epochs, and fine-tuning. The performance of SAM depends entirely on the performance of the detectors since it is a tracker following the detection path.

In future work, the system can be trained for multiple classes (multiple object types) as it can be used for different locations in video and different objects can be detected and classified. Real-time detection in Live matches and tracks the object and segments them. Segment Models can be applied and tested for the proposed object detection and segmentation and some unique results will be obtained which can be studied for analysis.

REFERENCES

[1] Z. Wang, B. Jiao and L. Xu, "Visual Object Detection: A Review," 2021 40th Chinese Control Conference (CCC), Shanghai, China, 2021, pp. 7106-7112, doi: 10.23919/CCC52363.2021.9550689. M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[2] Sharma, R., Saqib, M., Lin, C.T. et al. A Survey on Object Instance Segmentation. SN COMPUT. SCI. 3, 499 (2022). https://doi.org/10.1007/s42979-022-01407-3

[3] Joe R. Brown and Edward E. DeRouin "Integrated detection and segmentation for hyperspectral imagery using neural networks", Proc. SPIE 1965, Applications of Artificial Neural Networks IV, (2 September 1993); https://doi.org/10.1117/12.152574

[4] Alexander Kirillov, Eric Mintun, Nikhila Ravi1, Hanzi Mao2 Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, Ross Girshick "Segment Anything" (Apr 2023). https://doi.org/10.48550/arXiv.2304.02643

[5] "Performance metrics," towardsdatascience.com. [Online].Available:https://towardsdatascience.com/performance-metrics-for-classification-machine-learning problems-97e7e774a007.

[6] "Roboflow," Available: https://app.roboflow.com/.

[7] "Google Colaboratory,"Colab.research.google.com, 2019.[Online].Available:https://colab.research.google.com/.