IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Computer Vision Identification Using Custom Dataset

[1]Aliya Banu,[2]Bandu Bhai Mohammed Eissa,[3]Arjun Ambekar

[1]Student,[2]Student,[3]Faculty

[123]Department of Electronics and Communication Engineering,

[123]Kishkinda University, Ballari, India

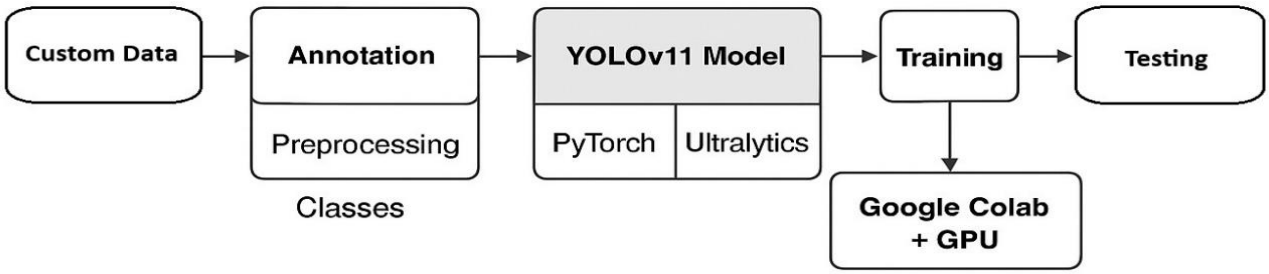
Abstract: This project explores the use of the YOLOv11 deep learning model for advanced object detection to identify and classify custom visual elements within images. The main goal is to develop a model capable of accurately detecting specific objects under diverse and challenging image conditions. To this end, a custom dataset was created and carefully annotated to represent the objects of interest. The training was performed on Google Colab, taking advantage of GPU acceleration to speed up computation. Data Preprocessing techniques such as image augmentation and normalization were applied to improve the model's robustness and ability to generalize across different scenarios. YOLOv11 was selected for its well-known balance between real-time processing capabilities and high detection accuracy, making it suitable for deployment on a range of devices from PCs to edge platforms. The trained model showed strong performance in detecting objects in varied environments and image qualities, demonstrating its practical applicability. This project highlights the effectiveness of modern object detection architectures in customized image analysis tasks and sets a foundation for future work, including real-time deployment and integration with embedded or mobile systems. The results indicate significant potential for automating visual recognition in security, automation, and intelligent system applications.

Index Terms - Real-time object detection, Custom dataset, Deep learning, Feature extraction, Bounding box localization, Mean Average Precision (mAP), Data augmentation, Image normalization.

I. INTRODUCTION

This project aims to develop a robust and efficient object detection system utilizing the YOLOv11 deep learning architecture, one of the latest advancements in the YOLO (You Only Look Once) family of models. Object detection plays a pivotal role in computer vision by enabling machines to not only identify objects within images and videos but also accurately localize them with bounding boxes. This capability is essential for a wide range of applications, including surveillance, autonomous driving, robotics, and smart city infrastructure. YOLOv11 is chosen for this project due to its superior balance between detection speed and accuracy, making it highly suitable for real-time applications on various platforms, from high-end GPUs to resource-constrained edge devices like Raspberry Pi. The goal is to harness YOLOv11's powerful feature extraction and detection capabilities to recognize and classify specific objects within a custom dataset tailored to the project's needs. The project begins with the careful collection of a comprehensive and diverse dataset, which involves gathering images that reflect the range of scenarios in which the object detection system will operate. Each image is meticulously annotated to mark the location and class of each object, ensuring high-quality training data. To improve the dataset's efficacy, various data preprocessing steps are employed, such as image resizing, augmentation (including rotation, flipping, and colour adjustments), and normalization, to increase the model's robustness and ability to generalize. Training the YOLOv11 model involves feeding the prepared dataset into the network, where it learns to detect and distinguish objects by analyzing patterns and features at multiple scales. Throughout training, hyperparameters are fine-tuned, and techniques such as transfer learning may be utilized to speed up convergence and improve model

performance. The training process is closely monitored, with validation datasets used to prevent overfitting and ensure the model's adaptability to new, unseen data. Once trained, the model undergoes comprehensive evaluation using established metrics such as precision, recall, mean Average Precision (mAP), and inference speed (frames per second). These evaluations assess not only the accuracy of object detection but also the practical viability of deploying the model in real-time environments. Additional tests under varying lighting, background clutter, and occlusion conditions validate the model's robustness.



II. METHODOLOGY

This project follows a systematic approach to develop an object detection system using the YOLOv11 architecture. The methodology is divided into several key stages: dataset preparation, data annotation, model training, model evaluation, and real-time object detection. Each stage is critical to ensure the overall performance and accuracy of the detection system.

1. DATASET PREPARATION:

The first step involves collecting a diverse set of images within our university environment containing the target objects: University logo, University title, and Photo of the founder. The images include various angles, lighting conditions, and backgrounds to improve the model's robustness, accuracy, and precision. The dataset is divided into a 90:10 ratio for training and testing purposes respectively.

2. DATA ANNOTATION:

The collected images are manually annotated using the Label Studio tool. Bounding boxes are drawn around each target object in batches, and labels are assigned accordingly. This annotated dataset forms the custom training data for the YOLOv11 model.

3. DATA PREPROCESSING:

Preprocessing steps include maintaining consistent image resolution and preparing the dataset to align with training requirements. This stage ensures data quality and suitability for efficient model training. Data augmentation techniques may be applied to improve model generalization.

4. MODEL TRAINING:

The YOLOv11 model is configured and trained on the pre-processed dataset. During training, model weights are optimized to minimize detection errors. Key hyper parameters such as learning rate, batch size, and number of epochs are carefully tuned to enhance performance**.**

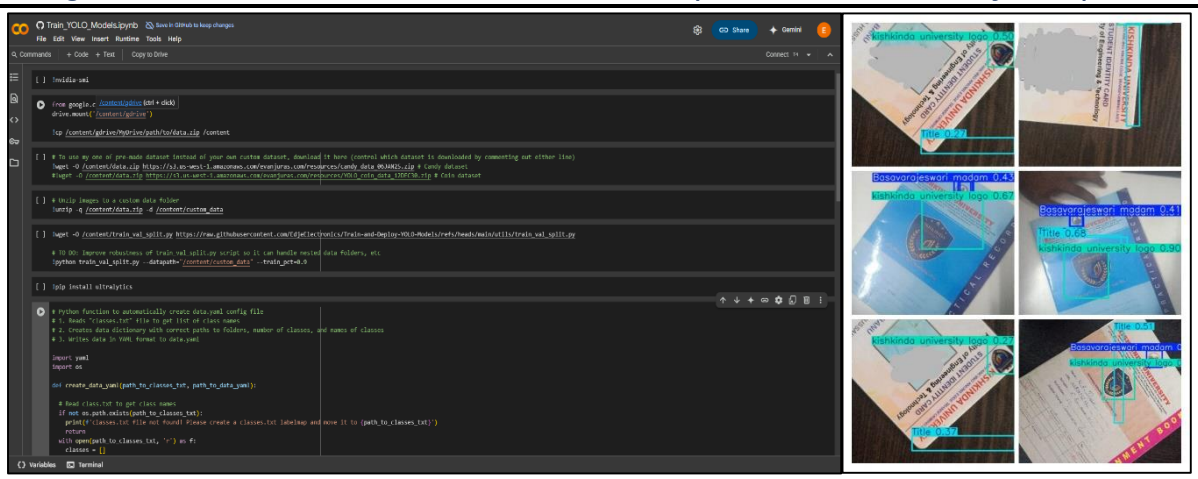


Figure: Simulation and Result

The figure consists of four images, each displaying the output of an object detection model trained using YOLOv11 to identify specific classes from Kishkinda university documents. Your model is detecting three custom classes: Kishkinda university logo Title Photo of a founder Top-Left Image (Student ID Card) Detected Objects: Kishkinda university logo with 0.83 confidence Title with 0.37 confidence Interpretation: The logo is accurately detected with high confidence.

The Title detection is visible but less confident, likely due to variation in font, size, or its position on the card. The image demonstrates successful detection on smaller and compact document layouts. Top-Right Image (Practical Record Book - Blue) Detected Objects: Kishkinda university logo – 0.67, Title – 0.33, Photo of a founder– 0.43, Interpretation: The model effectively identifies multiple objects.

The confidence for the logo is decent but lower than in the ID card. The Title and Photo of a founder. labels have lower confidence, indicating potential need for more annotated samples or improved class distinction. Bottom-Left Image (Same Practical Book with Hand) Detected Objects: Kishkinda university logo – 0.90, Title – 0.68 Photo of a founder. – 0.41, Interpretation: Excellent confidence in detecting the logo — the highest in all images. Despite a hand partially obstructing the book, the model still performs well, indicating good robustness

The Title class detection is stronger here, likely due to better visibility. Bottom-Right Image (Assignment or Lab Record Book - Pink Label) Detected Objects: Kishkinda university logo – 0.71, Title – 0.51Photo of a founder. – 0.54

III. RESULTS AND DISCUSSION

The results displayed in the figure represent the output of an object detection system developed using the YOLOv11 architecture, trained specifically to recognize visual elements from Kishkinda University documents. The model was designed to detect three custom classes: the university logo, the document title, and the image of “Basavarajeswari madam.” Training was conducted using the PyTorch-based Ultralytics implementation of YOLOv11, taking advantage of GPU acceleration on Google Colab for efficient processing. The dataset, consisting of university ID cards, practical record books, and various academic documents, was manually annotated using the Label Studio tool. Data preprocessing included normalization and resizing to maintain consistency across input samples. The resulting model successfully identified the defined classes, with high confidence values for the university logo, indicating strong recognition capability for well-defined, repeatable visual features. The detection of titles and the individual’s image, although accurate in location, exhibited slightly lower confidence scores—likely due to variability in appearance, orientation, or background context. Despite these challenges, the model demonstrated robust performance in diverse environments, handling partial occlusions, varied lighting conditions, and complex layouts with reasonable accuracy.