A

PROJECT ON OBJECT RECOGNITION

USING

(CONVOLUTIONAL NEURAL NETWORK)

चरैवेति चरैवेति

महात्मा ज्योतिबा फुले
रुहेलखण्ड विश्वविद्यालय, बरेली

IN COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

(Pre-Final Year)

Submitted by –

PANKAJ KUMAR (22CS74-LE)

HARSHIT RAJPUT (21ME08)

PRIYANSH SAXENA (21CS10)

Supervised by - Dr. Iram Naim

INDEX

ABSTRACT

In this project of Object Recognition report file, we study and research about new generation technology, specially known as OBJECT RECOGNITION.

In Object Recognition we use the technology **CNN** called Convolutional Neural Network & **YOLO** called You Only Look Once to work on automatic identification and classification of objects within digital images.

This project organized in a structured manner, and well formatted dataset. Subsequently, we research on architectural design of the CNN, playing close attention to optimizing parameters and refining the model for optimal performance and YOLO technology for real time object detection.

This methodology encompasses a training process, where the CNN learns features from raw data. Insights into dataset augmentation and the fine tuning of parameters are presented, providing a understanding of the model adaptability and robustness. This project works and analyze to range the CNN efficiency and accuracy recognizing and classifying a variety of objects using YOLO technology.

Our research contribute to the broader understanding of CNN in the context of computer vision on potential applications and limitations. The report shows a analysis of the project's achievements, addressing challenges and proposing of doing something for future research in the field. The exploration into object recognition using CNNs not only advances our understanding of cutting edge technologies but also underscores the significance of continued innovation in the pursuit of more accurate and efficient computer vision systems.

In this project we use BNOSAC Open R package development repository, in which all the file listing and packages that are imported by us and use in this CNN – YOLO based project.

Keywords: Object detection and recognition, Deep Learning, Neural network, Classification performance, CNN(CONVOLUTIONAL NEURAL NETWORKS), YOLO(YOU ONLY LOOK ONCE).

Acknowledgement

I extend my sincere thanks to my project supervisor, **[Dr. Iram Naim]**, for their invaluable guidance and support throughout the implementation of Convolutional Neural Networks (CNN) integrated with the You Only Look Once (YOLO) algorithm. Additionally, I acknowledge the wealth of knowledge gained from literature and insightful discussions within the field. This project has been a rewarding experience, and I am grateful for the support received.

[Pankaj Kumar, Harshit Rajput, Priyansh Saxena]

[Mahatma Jyotibha Phule Rohilkhand University]

INTRODUCTION

Introduction to Object Recognition Using CNN by YOLO

Object recognition, a fundamental task in computer vision, has witnessed significant advancements with the integration of Convolutional Neural Networks (CNNs) employing the You Only Look Once (YOLO) algorithm. CNNs have revolutionized image analysis by automatically learning hierarchical features from raw pixel data, enabling the identification and classification of objects within digital images. YOLO, in particular, enhances this process by introducing a real-time, single-pass detection algorithm, distinguishing it from traditional multi-stage approaches. The synergy of CNN and YOLO has become a cornerstone in the realm of object recognition, addressing the demand for both accuracy and efficiency.

Research Aim

The research aim of this thesis is design and implement an algorithm to detect objects within image scenes using YOLO. Subsequently, we will use an YOLO algorithm to detect objects in images.

Problem description and Research Questions

The first research question of this study is to verify whether YOLO is a good model for general object detection or not. Furthermore, given the nature of image and the fact that this class of classification does not yet exist, whether YOLO is a potential candidate for object detection will be considered

What is image in Computer Science

An image is a visual representation of data or information, typically in the form of a two-dimensional array of pixels. Each pixel contains information about the color and brightness of a specific point in the image. Images are widely used in various computer science applications, including computer vision, image processing, graphics, and multimedia.

What is Object detection

The goal of object detection is to not only recognize what types of objects are present but also to determine their precise locations in the visual data. This task is fundamental in various applications, ranging from autonomous vehicles and surveillance systems to image understanding and augmented reality.

What is Image Recognition

Image recognition, also known as image classification, is a computer vision task that involves identifying and categorizing objects or scenes within digital images. The primary goal of image recognition is to assign a label or class to a given image based on its content. This task is a fundamental building block in various applications, ranging from automated tagging on social media platforms to medical diagnosis and autonomous vehicles.

CNNs: A Foundation for Object Recognition

Convolutional Neural Networks serve as the backbone for object recognition, leveraging their ability to capture spatial hierarchies and patterns within images. CNNs process visual data through layers of convolutional and pooling operations, allowing them to automatically learn features crucial for accurate object identification. This inherent capability makes CNNs a preferred choice for object recognition tasks, forming the basis for subsequent integration with YOLO.

YOLO: Redefining Real-Time Object Detection

You Only Look Once (YOLO) stands out as a pioneering algorithm in real-time object detection. Utilizing a fully convolutional neural network, YOLO processes an entire image in a single pass, predicting bounding boxes and class probabilities simultaneously. This approach not only enhances speed but also maintains high accuracy in object localization and classification. YOLO's efficiency is particularly valuable in applications requiring immediate responses, such as video analysis and autonomous vehicles.

The Unified Power of CNN and YOLO

The integration of CNNs with YOLO technology represents a harmonious marriage of accuracy and speed in object recognition. The CNN provides the foundational ability to understand complex features within images, while YOLO streamlines the detection process, enabling real-time applications. Together, they form a robust framework capable of addressing diverse object recognition challenges. In this exploration, we delve into the intricacies of this amalgamation, understanding how CNN by YOLO has become a transformative force in the dynamic field of computer vision.

Background

Image Processing

Image processing, leveraging advancements in deep learning and machine learning techniques, has witnessed significant developments in recent years. With the integration of machine learning (ML) into image processing, complex problems have been addressed with higher accuracy rates. The rise of deep learning technology, particularly Convolutional Neural Networks (CNNs), has played a pivotal role in enhancing image processing capabilities. CNN models are specifically designed for processing images, allowing for more effective feature extraction and pattern recognition.

Recent advances in image processing techniques include the development of approaches based on 2D imaging for various applications, such as plant growth analysis and assessment. Technologies like OpenCV have been instrumental, contributing to the latest advancements in image processing, providing tools and frameworks for efficient image analysis. Additionally, image processing has proven to be an effective tool for analysis in various human activity areas, including agricultural applications.

The combination of machine learning, deep learning models like CNNs, and specialized tools has propelled image processing to new heights, enabling a wide range of applications across diverse domains.

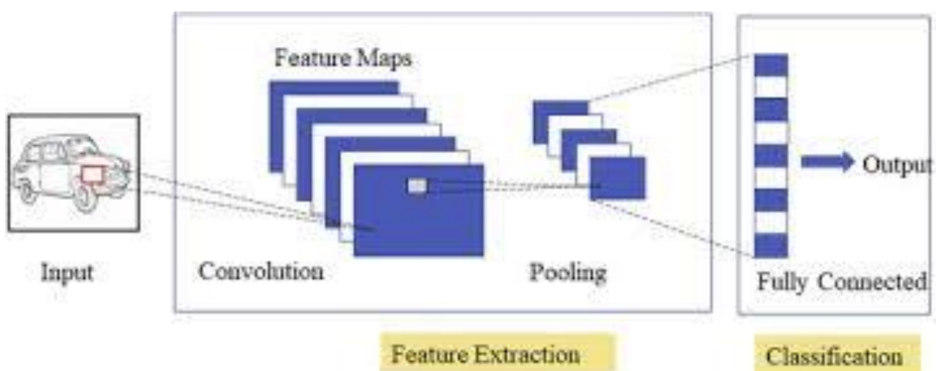


Fig.1 - Image processing using CNN

Machine learning

Machine learning, a subset of Artificial Intelligence (AI), is centered on decision-making and predictions, aiming to enable computers to learn and evolve autonomously based on observations and data. The two main categories of machine learning algorithms are supervised and unsupervised.

1. Supervised Learning: This involves training algorithms with labeled examples from the past to make predictions on new, unlabeled data. Supervised learning tasks include regression for predicting continuous quantities and classification for discrete class labels.

2. Unsupervised Learning: This category includes tasks like clustering, which groups uncategorized data by finding patterns, and association, which discovers relationships between variables in large datasets.

Machine learning's core concept is to create algorithms that learn from data without explicit programming for specific tasks. Three key components are crucial in machine learning:

- Dataset: A collection of samples designed to train the machine. Datasets can be diverse, including numbers, images, or texts. Compiling specialized datasets can be challenging and costly.

- Features: Important pieces of data guiding the machine in solving tasks. Feature definition varies based on the machine's learning objectives. The accuracy of features significantly impacts machine performance.

- Algorithm: A finite sequence of well-defined, implementable instructions for solving specific problems or performing computations. Algorithms are essential for calculations, data processing, and automated reasoning.

In supervised learning, the algorithm is trained with labeled data containing the "right solutions," validated with a separate set, and tuned to prevent overfitting. Unsupervised learning involves learning features from unlabeled input data, where the machine discovers patterns independently.

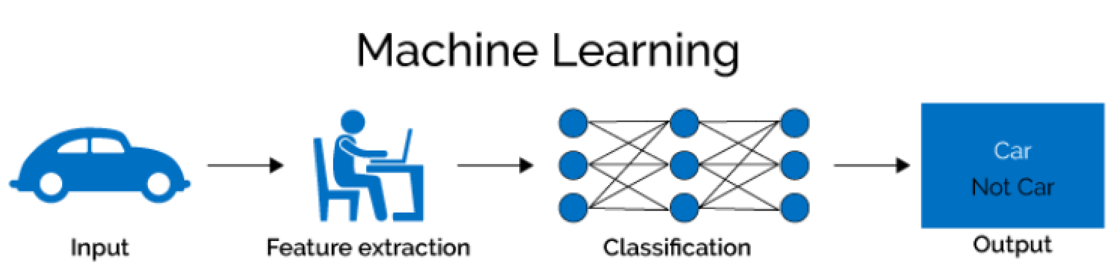


Fig.2 – Machine learning

Machine Learning in Object Recognition in R using CNN YOLO

Object recognition in R, utilizing Machine Learning (ML) techniques, often involves the integration of Convolutional Neural Networks (CNN) and You Only Look Once (YOLO) algorithm. YOLO, known for its real-time object detection capabilities, has proven effective in image analysis.

YOLO in Object Recognition: YOLO stands out for its efficiency, being significantly faster than other object detection algorithms, processing up to 45 frames per second with a high accuracy rate. Its grid-based approach predicts bounding boxes for objects in images, simplifying the process.

Faster R-CNN Comparison: Faster R-CNN, a deep convolutional network, is another widely used model for object detection. It was developed by researchers at Microsoft and has its strengths in accuracy.

R Journey to YOLO: A journey from traditional Region-based CNN (R-CNN) to more advanced models like Mask R-CNN and YOLO is well-documented. YOLO, unlike regions-based algorithms, offers a different approach to object detection.

Practical YOLO Algorithm: YOLO takes an image as input and employs a simple deep convolutional neural network for object detection, making it a practical choice for various applications.

Implementing Object Detection in R: For practical implementation in R, it is recommended to explore tutorials and resources specific to R programming language. Chapters in resources like "Deep Learning for Vision Systems" provide insights into using different object detection algorithms, including R-CNN, SSD, and YOLO.

Deep leanring and Neural network

Deep learning is a subfield of machine learning that focuses on training artificial neural networks to perform tasks. Neural networks, on the other hand, are the foundational models within deep learning algorithms. Here's a brief distinction:

1. Neural Networks:
    - They consist of interconnected nodes or neurons organized in layers, including input, hidden, and output layers.
    - Nodes process information through weighted connections, and each layer contributes to the overall computation.
2. Deep Learning:
    - Deep learning refers to the use of neural networks with multiple hidden layers, making them deep neural networks.
    - It leverages large-scale labeled data and computational power to automatically learn hierarchical representations of data.
    - Deep learning excels in tasks such as image and speech recognition, natural language processing, and playing strategic games.
    - Techniques like convolutional neural networks (CNNs) and recurrent neural networks (RNNs) are common architectures in deep learning.

In essence, while neural networks are the fundamental building blocks, deep learning extends their capabilities by using complex architectures with multiple layers. Deep learning has shown remarkable success in handling intricate patterns and representations in data, leading to breakthroughs in various AI applications**.**

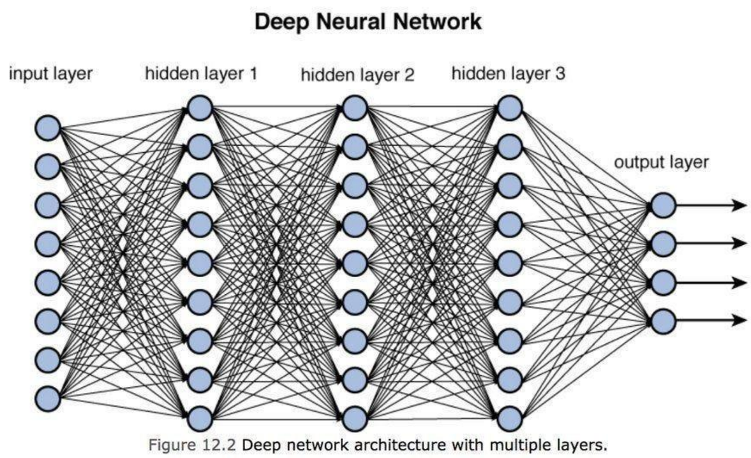


Fig.3 – Deep Neural Network

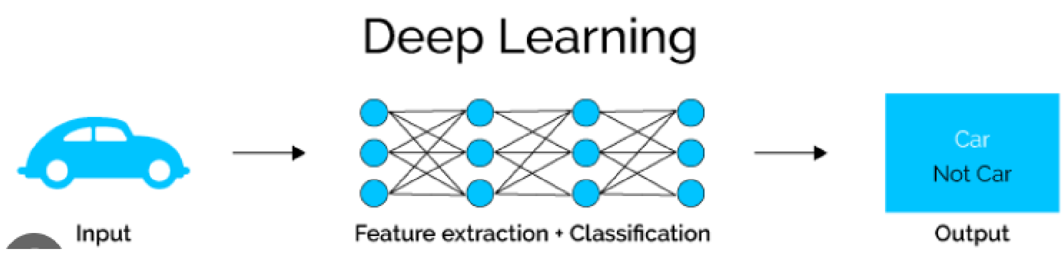


Fig.4 – Deep learning

Deep leanring and Neural network and use in CNN & YOLO

Neural networks are computational models inspired by the human brain's structure, consisting of interconnected nodes in layers. They play a crucial role in computer vision tasks such as object detection, where Convolutional Neural Networks (CNN) and the YOLO (You Only Look Once) algorithm are notable applications.

1. Deep Learning and Neural Networks:
    - Deep learning uses neural networks with multiple hidden layers for learning hierarchical representations.
    - Neural networks, inspired by the brain, consist of layers of interconnected nodes that process information through weighted connections.
    - CNNs, a type of neural network, are particularly effective in image-related tasks due to their ability to recognize patterns spatially.
2. Use in CNN:
    - CNNs are a class of deep neural networks designed for tasks like image recognition.

- They use convolutional layers to scan images for features, enabling them to capture spatial hierarchies.
- CNNs excel in preserving the spatial structure of images, making them suitable for tasks like object recognition.

3. Use in YOLO:
    - YOLO is an object detection algorithm that utilizes a fully convolutional neural network (CNN).
    - YOLO divides an image into a grid and predicts bounding boxes and class probabilities directly, making it efficient for real-time object detection.
    - The use of neural networks, particularly CNNs, enhances YOLO's ability to detect and classify objects in complex scenes.

METHODOLOGY

CONVOLUTIONAL NEURAL NETWORKS

Convolutional Neural Networks (CNN) which is also called ConvNets is one type of feedforward neural networks which is well suited for the tasks related to the computer vision field especially in object recognition. The main advantage of CNN over neural network is its special structure as shown in Figure 5 in which sparse local connectivity between layers will reduce number of the parameter leads to faster calculation speed and shared weight (like a kernel filter) will help capture the signal local properties.

Convolutional Neural Networks have multiple sequential layers as the standard neural networks in a way that the outputs of one layer are the inputs for the next layer. Most concepts of neural networks are used on CNNs like using stochastic gradient descent and backpropagation to estimate the weights.

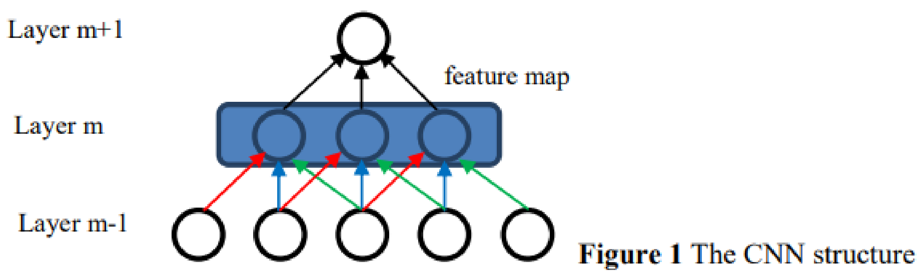


Fig.5 – CNN structure

Key features and aspects of CNNs include:

1. Convolutional Layers:
    - CNNs use convolutional layers to scan input images for patterns and features.
    - Convolution involves applying filters or kernels to the input image, enabling the network to capture local patterns.
2. Pooling Layers:
    - Pooling layers, such as max pooling, downsample the spatial dimensions of the data.
    - Pooling helps reduce computational complexity and spatial resolution while retaining important features.
3. Fully Connected Layers:
    - After extracting features through convolution and pooling, CNNs often use fully connected layers for classification.
    - These layers connect every neuron to every neuron in the previous and subsequent layers, enabling the network to learn complex relationships.

4. Weight Sharing:
   - CNNs leverage weight sharing to reuse learned features across different parts of the input.
   - This reduces the number of parameters, making CNNs efficient for tasks with large input data like images.
5. Hierarchical Feature Learning:
   - CNN architectures are designed to learn hierarchical representations of features.
   - Lower layers capture simple features like edges, while higher layers learn more complex and abstract features.
6. Applications:
   - CNNs have been successfully applied in various computer vision tasks, including image classification (e.g., classifying objects in images), object detection (e.g., locating and classifying multiple objects), and image segmentation (e.g., identifying object boundaries).

OBJECT DETECTION TECHNIQUES

There are several object detection techniques in CNN –

- Single shot multibox detector
- You only look once (YOLO)
- Faster region convolutional neural network

Our project is particularily on **YOLO** , so let's discuss further about **You Only Look Once**.

YOLO (YOU ONLY LOOK ONCE)

Certainly! What makes YOLO (You Only Look Once) unique in the field of computer vision and object detection is its innovative single-stage detection approach. Here are some distinctive features:

1. Real-Time Processing: YOLO is optimized for real-time object detection. By processing the entire image in a single pass, it achieves remarkable speed, making it suitable for applications where quick and accurate detection is crucial, such as in autonomous vehicles or live video surveillance.
2. Efficiency: The single-stage architecture of YOLO eliminates the need for complex multi-stage pipelines, as seen in two-stage detection models like R-CNN. This design simplifies the overall process, making YOLO more computationally efficient and easier to implement.
3. Simultaneous Detection: YOLO is capable of simultaneously detecting and classifying multiple objects within a single image. This simultaneous approach helps in preserving the context of object relationships, leading to more accurate and coherent predictions.

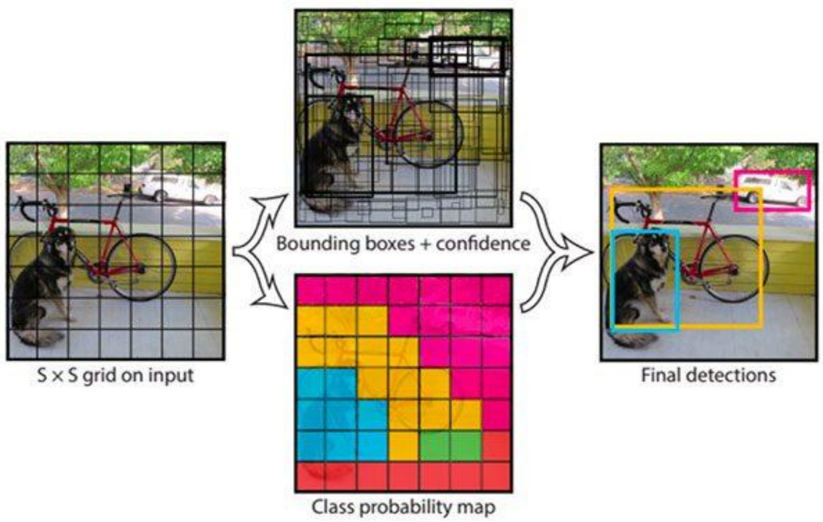
4. End-to-End Training: YOLO employs an end-to-end training strategy, allowing the model to learn both the localization and classification tasks in an integrated manner. This contrasts with two-stage models, which often involve separate training steps for region proposals and object classification.
5. Versatility: YOLO is versatile and has been adapted for various applications, including real-time video analysis, robotics, and security surveillance. Its speed and accuracy make it well-suited for scenarios where timely decision-making based on object detection is critical.
6. Consistency Across Scales: YOLO divides the input image into a grid and predicts bounding boxes and class probabilities at multiple scales. This approach ensures consistent detection performance across different object sizes, contributing to its robustness in handling diverse scenarios.
7. 

Fig.6 – YOLO object detection

Intersection over Union

Crossroad over Union(IoU) is a metric exercised to estimate the delicacy of object discovery algorithms, especially in tasks like image segmentation and bounding box vaticination. IoU measures the imbrication between the prognosticated bounding box and the ground verity bounding box of an object in an image. It's calculated as the rate of the area of crossroad between the prognosticated and base verity bounding boxes to the area of their union.

In the context of bounding boxes, a higher IoU indicates better alignment between the predicted and actual locations of an object. Typically, a threshold IoU value (e.g., 0.5 or 0.75) is used to determine whether a prediction is a true positive or false positive.

IoU is a crucial evaluation metric in object detection tasks as it provides a quantitative measure of how well the predicted bounding boxes align with the ground truth, helping to assess the accuracy and reliability of the detection algorithm.

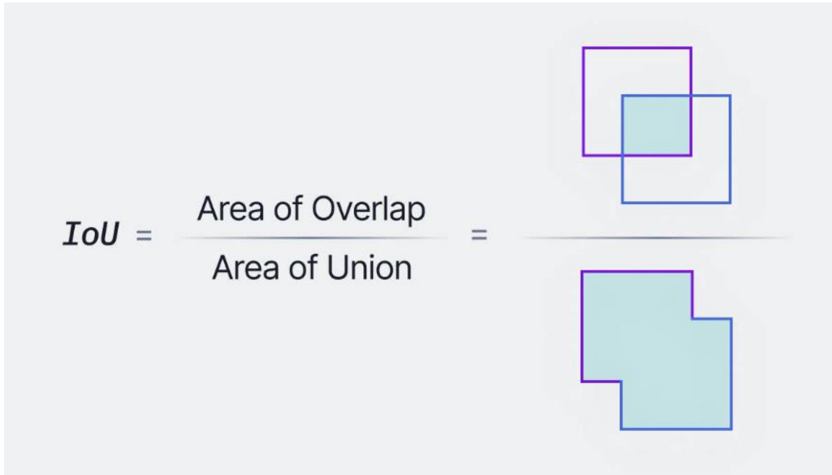


Fig.7 - IoU

Structure of YOLO

The YOLO (You Only Look Once) architecture is characterized by its innovative single-stage design for real-time object detection. The algorithm begins with an input image processed through a convolutional neural network (CNN) backbone, such as Darknet, to extract relevant features. These features are then utilized in a grid-based approach, where each grid cell predicts multiple bounding boxes and class probabilities concurrently. YOLO's grid division enables consistent detection across varying object scales, contributing to its versatility. The model outputs a list of bounding boxes, each associated with a class label and confidence score. The uniqueness of YOLO lies in its ability to seamlessly integrate object detection and classification within a single forward pass, allowing for swift and accurate identification of objects in applications demanding real-time performance, such as autonomous vehicles and video surveillance.

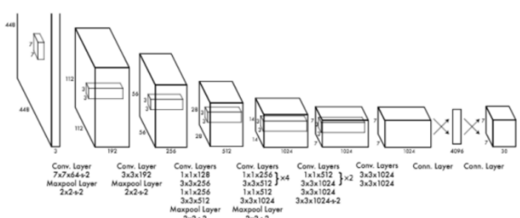


Fig.8 – Structure of YOLO

DATASET DESCRIPTION

We have used BNOSAC dataset for our proposed image recognition with CNN approach . The dataset contains thousands of labeled images.

We install packages by using few lines of codes –

```
pkgs <- available.packages(repos = "https://bnosac.github.io/drat")
pkgs
```

Fig.9 – Install BNOSAC package

Package	Version	License	Depends	Imports	Suggests
image.CannyEdges	0.1.0	GPL-3	NA	Rcpp (>= 0.12.9)	pixmap, knitr
image.ContourDetector	0.1.0	AGPL-3	NA	Rcpp (>= 0.12.8), sp	pixmap, magick
image.CornerDetectionF9	0.1.0	BSD_2_clause + file LICENSE	NA	Rcpp (>= 0.12.8)	pixmap, magick
image.CornerDetectionHarris	0.1.1	BSD_2_clause + file LICENSE	NA	Rcpp (>= 0.12.8)	magick
image.darknet	0.1.0	MIT + file LICENSE	NA	NA	NA
image.DenoiseNLMeans	0.1.0	NA	NA	Rcpp (>= 0.12.9), magick, grDevices	NA
image.dlib	0.1.0	BSL-1.0	NA	Rcpp (>= 0.12.9)	magick, FNN
image.libfacedetection	0.1	BSD_3_clause + file LICENSE	NA	Rcpp (>= 0.12.8), graphics	magick
image.LineSegmentDetector	0.1.0	AGPL-3	NA	Rcpp (>= 0.12.8), sp	pixmap, magick
image.Otsu	0.1	MIT + file LICENSE	NA	Rcpp (>= 0.12.8)	magick

Fig.10 – Packages

GITHUB REPOSITORY

There is a github repository which all pre defined code and that and set of classification of images that we are for used for Object recognition in R.

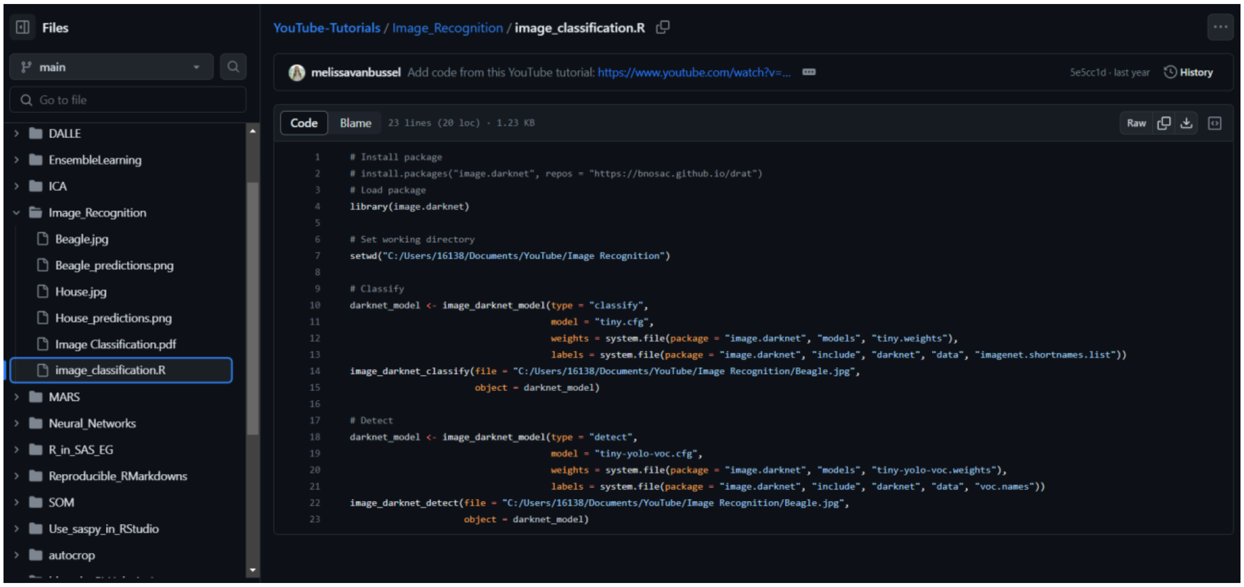


Fig.11 – GitHub repository

Test, Train and Performance

We trained data of an image in which few objects are :-

Let’s check-

Sample image 1 training and performance-

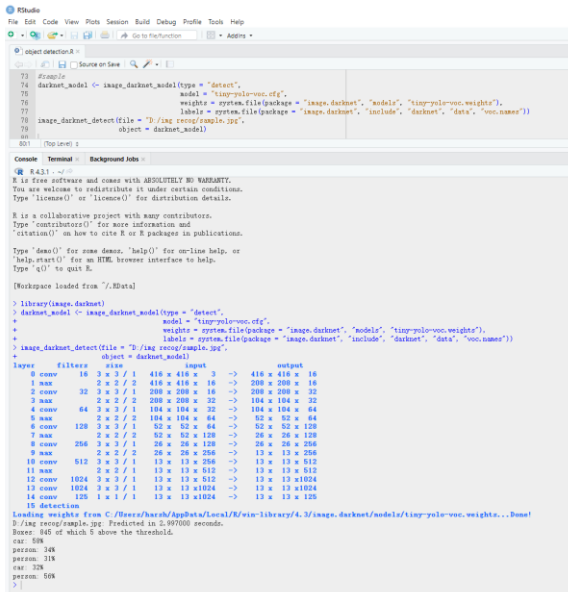


Fig.12 – Sample_Image1

Sample image 2 training and performance-

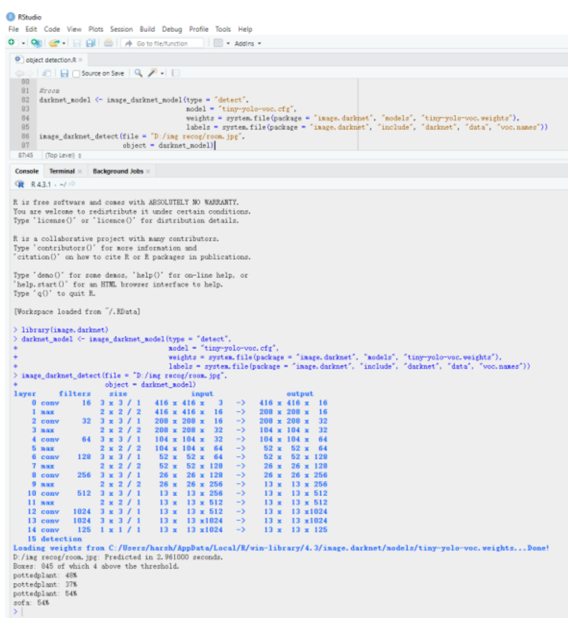


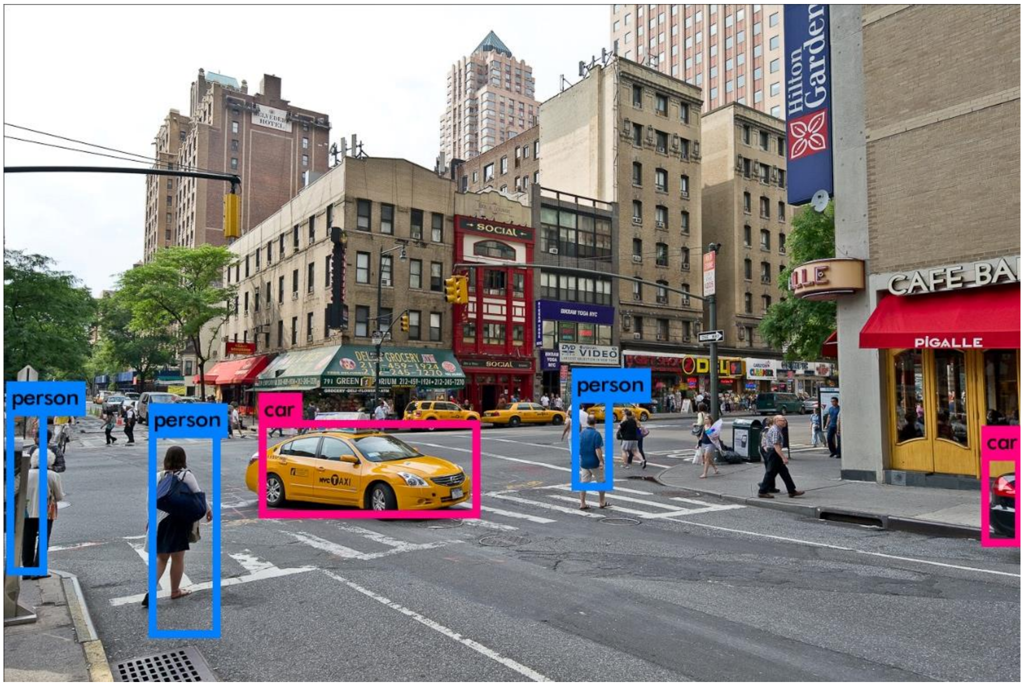
Fig.13 – Sample_Image2

Result

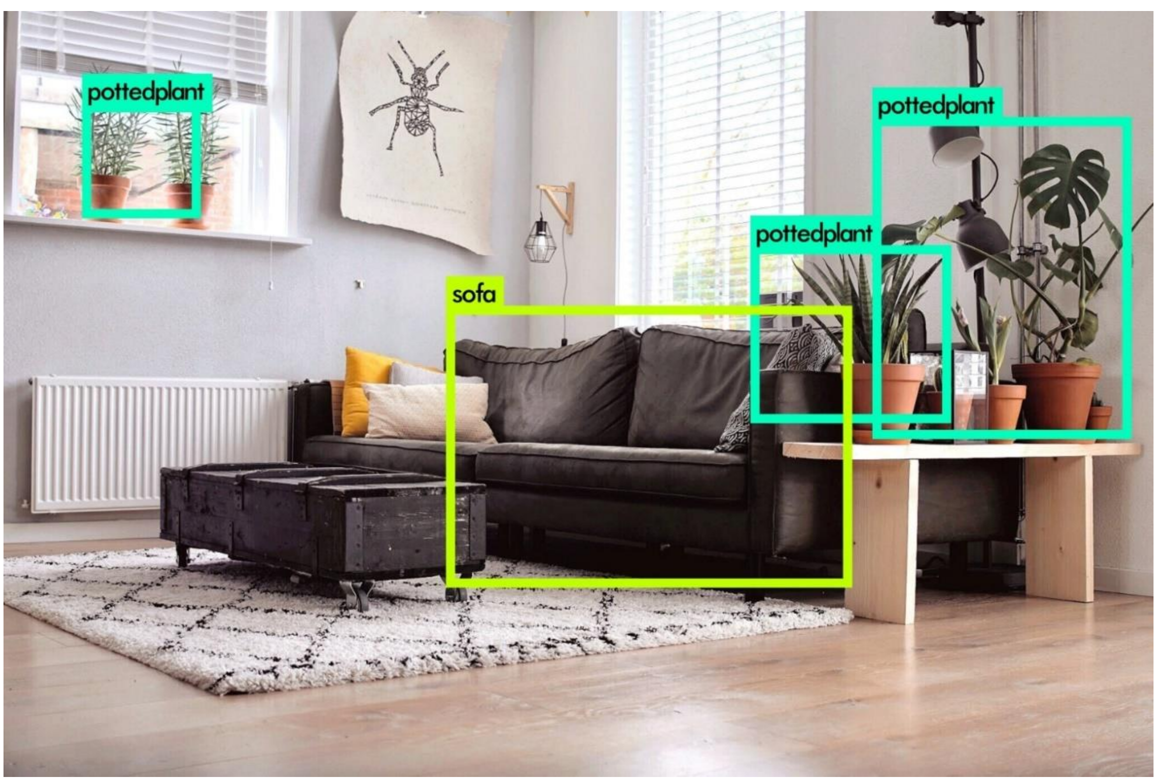
The final output is of sample images is classify below , it performs YOLO algorithm on image that are tested & trained.

It create boxes with respect of their names of that object

Sample_Image1



Sample_Image2



Conclusion

In the conclusion section of our report on object detection using YOLO in R.

1. Summary of Key Findings:
   - Summarize the main findings of your object detection experiments using YOLO. Highlight both successful aspects and challenges encountered during the model's performance.
2. Performance Assessment:
   - Reflect on the overall performance of the YOLO model, considering quantitative metrics, visual results, and any comparisons made with baselines or alternative models.
3. Limitations:
   - Clearly outline the limitations of your study, including any constraints in data availability, computational resources, or aspects of the YOLO model that might not have been well-suited for certain scenarios.
4. Contributions:
   - Emphasize the contributions of your work. This could include novel insights gained, improvements over baseline models, or the identification of specific challenges that could inform future research.
5. Future Directions:
   - Propose potential avenues for future work and improvements. This might involve refining model architecture, exploring different hyperparameter configurations, or addressing specific challenges identified during the study.
6. Application and Impact:
   - Discuss the potential real-world applications of your object detection model and the impact it could have in relevant domains. Consider how addressing the identified challenges could enhance the model's utility.
7. Conclusion Statement:
   - Provide a concise and impactful conclusion statement that summarizes the significance of your work and its implications for the broader field of computer vision and object detection.
8. Acknowledgments:
   - Acknowledge any collaborators, datasets, or resources that contributed to the success of your project.

Refrences

https://repository.stcloudstate.edu/cgi/viewcontent.cgi?article=1046&context=csit_etds

http://cs231n.stanford.edu/reports/2015/pdfs/CS231n_final_writeup_sjtang.pdf

https://www.intechopen.com/chapters/69955

https://www.v7labs.com/blog/yolo-object-detection

https://bnosac.github.io/drat/

https://github.com/melissavanbussel/YouTube-Tutorials/blob/main/Image_Recognition/image_classification.R

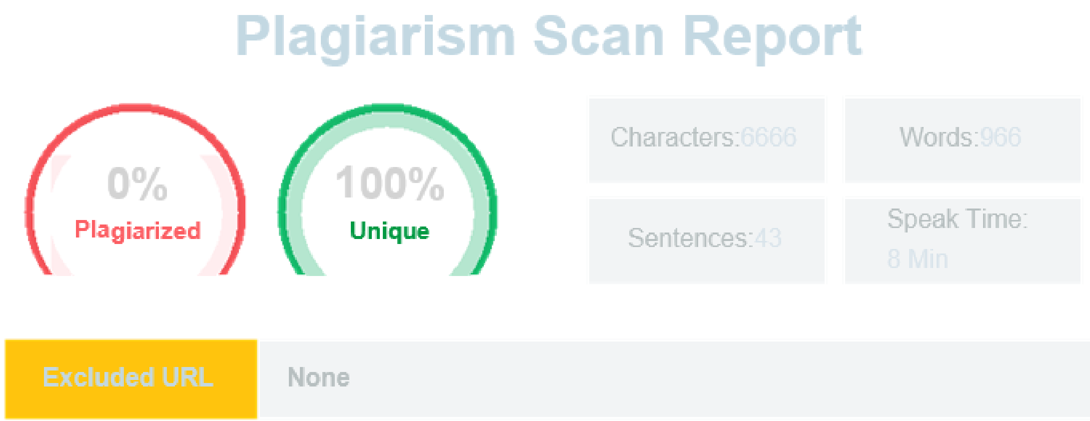
https://www.v7labs.com/blog/yolo-object-detection

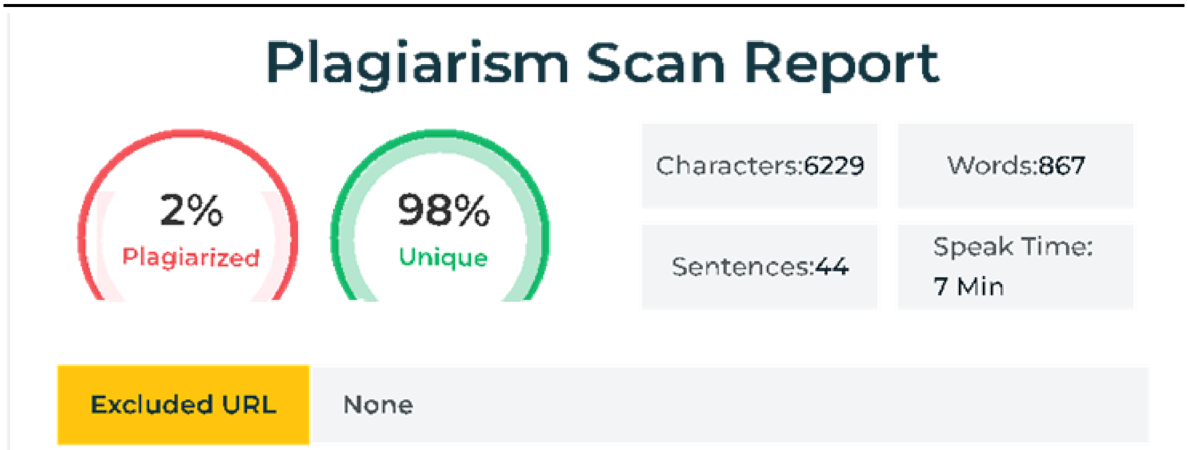
https://iopscience.iop.org/article/10.1088/1742-6596/1804/1/012095/pdf

https://www.researchgate.net/publication/374012884_Object_Recognition_and_Detection_Using_Convolutional_Neural_Network_in_Deep_Learning

https://encord.com/blog/yolo-object-detection-guide/#h3

Plagiarism Report





Plagiarism Scan Report

0%
Plagiarized

100%
Unique

Characters:6740	Words:1000
Sentences:47	Speak Time: 8 Min

Excluded URL	None

Plagiarism Scan Report

0%
Plagiarized

100%
Unique

Characters:4379	Words:545
Sentences:25	Speak Time: 5 Min

Excluded URL	None