

**INTELLIGENT HOMICIDE INVESTIGATOR:
A UNIQUE HOMICIDE CRIME SCENE INVESTIGATION AND DATA
COLLECTION TOOL USING CONVOLUTIONAL NEURAL NETWORK**

इंटेलिजेंट होमिसाइड जांचकर्ता: कन्वोल्यूशनल न्यूरल नेटवर्क का उपयोग करते हुए
एक अनोखा हत्याकांड अपराध स्थल की जांच और डेटा संग्रह उपकरण

**A
Thesis**

**Submitted for the Award of the Ph.D. degree of
PACIFIC ACADEMY OF HIGHER
EDUCATION AND RESEARCH UNIVERSITY**

By

NEHA VORA

नेहा वोरा

Under the supervision of

Dr. DIVYA SHEKHAWAT

Assistant Professor
Pacific Academy of Higher
Education & Research University, Udaipur



**FACULTY OF COMPUTER SCIENCE
PACIFIC ACADEMY OF HIGHER EDUCATION
AND RESEARCH UNIVERSITY, UDAIPUR**

2024

DECLARATION

I, **NEHA VORA D/O SHRI HANSRAJ GUPTA** resident of E-501, Shiv Manthan CHS, Chincholi Bunder Road, Malad West, MUMBAI- 400064, hereby declare that the research work incorporated in the present thesis entitled **“Intelligent Homicide Investigator: A Unique Homicide Crime Scene Investigation and Data Collection Tool Using Convolutional Neural Network”** (इंटेलिजेंट होमिसाइड जांचकर्ता: कन्वोल्यूशनल न्यूरल नेटवर्क का उपयोग करते हुए एक अनोखा हत्याकांड अपराध स्थल की जांच और डेटा संग्रह उपकरण) is my original work. This work (in part or in full) has not been submitted to any University for the award of a Degree or a Diploma. I have properly acknowledged the material collected from secondary sources wherever required.

I solely own the responsibility for the originality of the entire content.

Signature of the Candidate

Date:

FACULTY OF COMPUTER SCIENCE
PACIFIC ACADEMY OF HIGHER EDUCATION AND
RESEARCH UNIVERSITY, UDAIPUR

Dr. DIVYA SHEKHAWAT
Assistant Professor

CERTIFICATE

It gives me immense pleasure in certifying that the thesis **“Intelligent Homicide Investigator: A Unique Homicide Crime Scene Investigation and Data Collection Tool Using Convolutional Neural Network”** (इंटेलिजेंट होमिसाइड जांचकर्ता: कन्वोल्यूशनल न्यूरल नेटवर्क का उपयोग करते हुए एक अनोखा हत्याकांड अपराध स्थल की जांच और डेटा संग्रह उपकरण) and submitted by **NEHA VORA** is based on the research work carried out under my guidance. He / she have completed the following requirements as per Ph.D. regulations of the University;

- (i) Course work as per the University rules.
- (ii) Residential requirements of the University.
- (iii) Regularly presented Half Yearly Progress Report as prescribed by the University.
- (iv) Published / accepted minimum of two research paper in a refereed research journal.

I recommend the submission of thesis as prescribed/notified by the University.

Date:

Name and Designation of Supervisor

Dr. DIVYA SHEKHAWAT

Assistant Professor,
Pacific Academy of Higher Education
& Research University, Udaipur

COPYRIGHT

I, **NEHA VORA**, hereby declare that the Pacific Academy of Higher Education and Research University, Udaipur, Rajasthan, shall have the rights to preserve, use and disseminate this dissertation entitled **“Intelligent Homicide Investigator: A Unique Homicide Crime Scene Investigation and Data Collection Tool Using Convolutional Neural Network”** (इंटेलिजेंट होमिसाइड जांचकर्ता: कन्वोल्यूशनल न्यूरल नेटवर्क का उपयोग करते हुए एक अनोखा हत्याकांड अपराध स्थल की जांच और डेटा संग्रह उपकरण) in print or in electronic format for the academic research.

Date:

Signature of Candidate

Place:

ACKNOWLEDGEMENT

No great work is ever achieved in isolation, and this doctoral journey was no exception. The completion of my dissertation would not have been possible without the unwavering support of many remarkable individuals, to whom I owe a deep debt of gratitude. This acknowledgment is a humble tribute to them all.

First and foremost, I would like to extend my heartfelt appreciation to my guide, **Dr. Divya Shekhawat**. Her consistent support and encouragement, despite her numerous responsibilities, have been invaluable. Her belief in me and her inspiring guidance have played a crucial role in the successful completion of my thesis. It has been an honour and a privilege to work under her mentorship. Thank you, ma'am, for your constant help and encouragement.

I am also immensely grateful to the learned authors whose works provided the foundation for my research. Their contributions in the field of Object Detection have been instrumental in shaping this dissertation, and I deeply acknowledge their insights.

My sincere thanks go to the staff of Pacific Academy of Higher Education and Research University and U.P.G. College of Arts, Science & Commerce, for their invaluable assistance and motivation throughout this process.

I owe a special note of gratitude to my parents, **Hansraj Gupta** and **Rajbala Gupta**, for their blessings, and my sister-in-law, **Ankita Vora**, my in-laws, **Leela Vora** and **Bharat Vora**, and my son, **Krishnav**, for their unconditional support, patience, and love. Their faith in me has been a constant source of strength.


Special thanks to my husband **Mehul Vora**, for constantly supporting, encouraging and motivating me throughout this journey and my entire life.

Lastly, I express my deepest thanks to the Almighty for His grace and blessings, which have guided me through this journey.

I would like to thank each and every one who have helped me in my study, throughout the period. Last but not the least, my distinctive thanks to *Nav Nimantran Thesis Printing & Binding, Udaipur*, for their role in shaping the matter, creative design work and bringing out this document meticulously, neatly and timely.

DATE: -

NEHA VORA



DEDICATED TO
MY FAMILY, FRIENDS
AND WELL-WISHERS

PREFACE

The complex nature of crime scene investigations, particularly in cases involving homicides, has always presented numerous challenges to law enforcement agencies worldwide. From scattered physical evidence to the need for accurate victim identification, investigators are often faced with the monumental task of piecing together crucial details from what appears to be a chaotic scene. In such situations, forensic photography plays a vital role, allowing investigators to document the scene visually and analyze the crime from multiple perspectives. Yet, despite the importance of these photographs, human limitations frequently impede the investigation process. Even the most experienced investigators can overlook key details or struggle to process the vast number of images needed to form a coherent narrative.

This is where the inspiration for this research stems from. The traditional methods of documenting and analyzing crime scenes, while essential, are time-consuming and prone to human error. Recognizing the growing importance of technology, I became fascinated with the potential of artificial intelligence (AI) to transform the way forensic investigations are conducted. The idea of utilizing object detection and facial recognition technologies to assist investigators seemed not only plausible but essential in today's rapidly digitizing world.

In this work, I propose an intelligent evidence detection and collection system using a convolutional neural network (CNN) architecture, which aims to reduce the burden on forensic teams by automating the process of crime scene analysis. By detecting critical objects such as weapons, bottles, knives, etc and by identifying victims through facial recognition, this system is designed to accelerate investigations and improve their accuracy. The ultimate goal of this research is to provide law enforcement with a tool that can aid in early identification of crucial evidence, help in reconstructing the sequence of events at a crime scene, and ensure that justice is served with greater efficiency.

The scope of this study goes beyond the technical aspects of forensic science. It delves into the human element—the limitations we face in complex investigations and

the ways technology can bridge these gaps. The proposed system not only addresses the time-consuming process of manually reviewing images and making logs, but also ensures that no detail, no matter how minute, is missed during analysis. In particular, this model has the potential to revolutionize homicide investigations, where the early identification of a victim can be the key to solving the case swiftly. This research is the culmination of a deep-seated curiosity about the intersections of technology and criminal justice. It reflects years of study, experimentation, and a dedication to finding innovative solutions to long-standing problems in forensic science. It has been a journey shaped by both technical challenges and the invaluable support of my mentors, colleagues, and peers. Their guidance and insights have been instrumental in the development of this model, and I owe them a great debt of gratitude.

I also wish to acknowledge the profound impact of recent advancements in AI and machine learning, without which this work would not have been possible. Technologies like You Only Look Once (YOLO) for object detection and convolutional neural networks have opened new doors in the field of image analysis, and this research seeks to build upon these foundations to create a system that can directly impact criminal investigations in real-world scenarios.

As I present this work, I am filled with optimism about the future of forensic science. The integration of AI into crime scene analysis not only offers a way to enhance the precision and speed of investigations but also promises to reduce the emotional and mental strain on investigators. I hope that this research will serve as a stepping stone for further innovations, encouraging continued exploration of how technology can augment human capabilities in solving complex crimes. It is with great pride and a sense of responsibility that I offer this contribution to the field, with the hope that it will help bring about more effective and efficient investigations, ultimately leading to swifter justice for victims and their families.

Neha Vora

Mumbai

INDEX

CHAPTER- I INTRODUCTION		1 – 8
1.1	Scope of the proposed study	3
1.2	Review of work already done on the subject	3
1.3	Research gaps identified in the proposed field of the Research	5
1.4	Objective of the proposed study	5
1.5	Research Methodology & Detailed research plan	6
1.6	Chapter Scheme	7
CHAPTER- II REVIEW OF LITERATURE		9 - 19
CHAPTER-III FORENSIC PHOTOGRAPHY		20 – 39
3.1	The history of forensic photography	22
3.2	Information Fusion in Image Forensics	36
3.3	Foundations of Fuzzy Theory	37
CHAPTER- IV JOURNEY OF OBJECT DETECTION		40 – 55
CHAPTER- V EXPERIMENTAL SETUP AND EVALUATION		56 - 63
5.1	Advantages of Google Colab for Experimentation	56
5.2	Dataset Preparation	56
5.3	People Classes	59
5.4	Evaluation Metrics	62
CHAPTER- VI RESULTS AND DISCUSSION		64 - 106
6.1	YOLOV5 RESULTS	64
6.2	YOLOV7 RESULTS	72
6.3	YOLOV8 RESULTS	79
6.4	COMPARATIVE ANALYSIS OF YOLOV5, YOLOV7 AND YOLOV8 RESULTS	86
CHAPTER- VII CONCLUSION AND FUTURE WORK		107 – 111
7.1	Summary of Findings	107
7.2	Contributions and Implications of the Study	109
7.3	Future Research Directions in YOLO	110
7.4	Concluding Remarks	111
BIBLOGRAPHY		112 - 127
PUBLICATIONS		
CERTIFICATES		

LIST OF TABLE

Table No.	Particulars	Page No.
4.1	Summary of Improvements on YOLO	49
5.1	Class and no of images downloaded	57
5.2	Before & After Pre-processing People Image Dataset	60
5.3	Class instances and their no. of annotation	61
5.4	Dataset Summary	61
6.1	Result of YOLOV5 object detection model	64
6.2	Result of YOLOV7 object detection model	72
6.3	Result of YOLOV8 object detection model	79
6.4	Comparison of the various YOLO versions	86
6.5	mAP50 Performance of Individual class using YOLOv5, YOLOv7 and YOLOv8 models	87
6.6	mAP50-95: Performance of Individual class using YOLOv5, YOLOv7 and YOLOv8 models	88
6.7	Precision Performance of Individual class using YOLOv5, YOLOv7 and YOLOv8 models	89
6.8	Recall Performance of Individual class using YOLOv5, YOLOv7 and YOLOv8 models	91
6.9	F1 Score Performance of Individual class using YOLOv5, YOLOv7 and YOLOv8 models	92
7.1	Overall Performance	107
7.2	Class-Specific Performance	108
7.3	Recall	109

LIST OF FIGURE

Fig. No.	Particulars	Page No.
4.1	YOLO Architecture (Redmon et al., 2016)	48
4.2	YOLOv5 Architecture (Nepal & Eslamiat, 2022)	52
4.3	Model Scaling of YOLOv7 (Wang et al., 2022)	53
5.1	csv_folder and Dataset folder created	58
5.2	class-descriptions-boxable.csv and train-annotations-bbox.csv files created in csv_folder	58
5.3	class-descriptions-boxable.csv	59
6.1	Result of YOLOv5 model on Image	71
6.2	Result of YOLOv7 model on Image	79
6.3	Result of YOLOv8 model on Image	85
6.4	Precision -Confidence Curve of YOLOV5	94
6.5	Precision -Confidence Curve of YOLOV7	94
6.6	Precision -Confidence Curve of YOLOV8	95
6.7	Recall -Confidence Curve of YOLOV5	97
6.8	Recall -Confidence Curve of YOLOV7	97
6.9	Recall -Confidence Curve of YOLOV8	98
6.10	Precision- Recall Curve of YOLOV5	99
6.11	Precision- Recall Curve of YOLOV7	100
6.12	Precision- Recall Curve of YOLOV8	100
6.13	F1 Confidence Curve of YOLOV5	101
6.14	F1 Confidence Curve of YOLOV7	102
6.15	F1 Confidence Curve of YOLOV8	103
6.16	Confusion Matrix of YOLOV5	104
6.17	Confusion Matrix of YOLOV7	105
6.18	Confusion Matrix of YOLOV8	105

CHAPTER – I

INTRODUCTION



Crime scene is always a complex setting. There are evidences spread across the crime scene. It is very difficult for the investigator to remember the scene as it is. Here comes the role of the Forensic Photographer who clicks series of images to present a visual narrative of the crime scene. The pictorial documentation of the crime scene is important to ensure a thorough investigation. The investigator sees the images over and over again to make some sense of it. The investigator then connects the trail left in the photographs. Many theoretical, legal and technical problems come along with any photography, especially the forensic photography. As this is very technical in nature, any error or mistakes in the photography effects the overall investigation. A number of poorly shot and displayed photographs have the potential to sabotage the entire investigation. Therefore, crime scene photography is very important. The crime scene photography sets forth a visual storytelling of the crime that happened. As a basic guideline, the photographer goes from the general to specific. He clicks the entire crime scene from different angles and then moves towards any specific aspect or trail in the crime scene. As per the FBI's Forensic Science Training Manual on "The Fundamental Principles and Theory of Crime Scene Photography", the photographer covers the crime scene with three kinds of Shots

1. The Long range- that captures the entire crime scene and its settings, it can be from a vantage point or an Ariel View.
2. The mid-range -This illustrates many objects around the victim or the Scene.
3. Close Up shots- these are detailed shots of any mark, object etc.

There are then Follow Up photographs, Autopsy photographs and photographs of a live victim or suspect that display bruises or wounds. The most important aspect is the photography of the Physical evidence at the crime scene. There may be broken glass, flowerpot, Sofa that has stains, etc. These components establish the chain of custody of items presented in the courtroom during the trials.

Due to large number of photographed clicked at the crime scene, they need to be arranged in chronological manner to determine the chain of events. It is very important to make a Log. This Log contains the entire record of the photographs captured at the crime scene. It is very time consuming for the photographer to prepare the Log or report. Also, it is equally tedious to read the Logs.

Now for the investigating officer to remember each and every object in the photographs or video are always a challenge. He has to look at the images multiple times or to read the Logs. This results in reduced efficiency. It certainly will be a boon for the investigators when a Machine would analyse the images and give a list of objects detected. This will give the investigators an edge in solving crime. This would save the time of the photographer as well as the investigating officer.

Most importantly out of many crime scenes, homicide is always a challenge. In Homicide, there are many evidences that are spread across the crime scene that a human eye may often miss. There may be broken Knife, Handguns, Bottles, Axes, etc., or any such objects that may be relevant but does not catch any attention. In homicide crime scene there may be a resistance by the victim or a display of brute force by the attacker that may disrupt the setting of the location. If a machine could identify the objects in the crime scene and prepare a report of the same there could be lot of time saving in the investigation. The investigator would not need to make multiple trips to the crime scene or look at the photographs and review them multiple times.

Homicide crime scene analysis has always been a humongous task. It certainly demands the investigator to assess voluminous images. Another challenge of a homicide is timely identification of the victim. It is found that the attackers destroy any sign of identification of the victim they take away any ids, papers, phone and wallet that could help the law enforcement with investigation. Most of the time they dismember the body in faith to cause hurdles in investigation. Therefore, an early identification is essential to crack the case faster. It would be really nice if the machine could pull out every detail of a person thus by recognizing its face.

In early days this just seem hopeful but today in the age of digitization and advanced technologies such as artificial intelligence it is very much possible to do so.

The application of object detection and facial recognition would impact greatly on the time and efficiency of the investigation. The proposed model is trained to look at the aspects of object detection in crime scene and early victim identification.

The proposed model is aimed to overcome human limitations of taking a note of every object that is seen in the images it processes. So, that early investigation is carried and justice is served.

Understanding the benefits of Object detection in Crime scene analysis, we propose an architecture for intelligent evidence detection and collection system using convolutional neural network.

A sample data set is procured to train the model to detect the possible evidences and objects. The model also runs a facial recognition to identify the victim and pull out details to help the investigator. This proposed system aims to reduce the human efforts and time taken to scan through the crime scene for evidences; instead, it gives the name of objects such as gun, knife etc. from the video or image of the crime scene. This also saves multiple trips to the location and brings minute details of the crime scene to the attention of the investigator. The early identification helps in understanding the motif, thus giving an edge to the investigation and increased efficiency.

1.1 Scope of the proposed study

The scope of proposed research work is as below:

- A comprehensive model can be developed to detect crime scene objects.
- The proposed model also carries a facial recognition for early identification of the victim.
- The present study will increase the efficacy and accuracy in solving the crime.
- The proposed study will also save time and resources in solving a homicide crime.
- The proposed system display the objects detected which saves time.

1.2 Review of work already done on the subject

Previous Related Work

Traditionally Photography and videography are used to collect visual data from the crime scene. (Seckiner, Mallett, Roux, Meuwly, & Maynard, April 2018) Today, there are many technologies that ease the process of collecting, analysing and process the data collected. Today most phone has good resolution camera that makes photographing a crime scene easy, and with technologies such as artificial intelligence, machine learning and object recognition the processing of the data and

the utilization of information has significantly improved the way a crime scene is looked at. Taking this idea ahead Kamenicky *et al.* introduced various video methods and tools for processing digital images and videos for criminal investigation (Kamenicky *et al.*, July 2016). However, the researcher faced challenges to identify the source of the video and images. Many investigators today prefer to use their phones for recording videos and photos as its very handy and flexible. However, many phones produce noise to which Li (Li, Maa, & Wanga, November 2018) proposed an algorithm to extract Photo Response Nonuniformity (PRNU) noise from video that is shot by phone camera.

Today CCTV systems contribute as source of large amount of Visual data that is stored every day. These hidden surveillance cameras often record the whole crime right from its inception. CCTV footage plays Key role in digital forensics. The strength of the evidence relies on the visuals it has recorded (Seckiner, Mallett, Roux, Meuwly, & Maynard, April 2018). Therefore, the quality of the footage is very vital for the contents-based proofs recorded. Jenkins and Kerr (Jenkins & Kerr, 2013) proposed an images analysis using facial recognition that can be implemented on high quality recording devices, like smart mobile phones for better investigation. Today this automated image analysis from video is used for public security or detection of dangerous circumstances in many places around the world (Grega, Mاتیolański, Guzik, & Leszczuk, 2016).

Typically, CNN consist of multiple convolution layers, ReLU (Rectified Linear Units), pooling layers and fully connected layers. The last layers of a CNN generate activations that act descriptor for object detection and classification. Babenko (Babenko, Slesarev, Chigorin, & Lempitsky, 2014) named them as neural codes and proved that such activations can be used for image retrieval task. They also found that the descriptors perform competitively even if a CNN is trained for separate classification task. For instance, the objects in Imagenet (Russakovsky, Deng, Su, & al, 2015) datasets and MS-COCO (Lin, et al.) datasets can be used interchangeably for training the CNN model. Soon, Girshik (Girshick, 2015) proposed Fast R-CNN, that replaced SVM classifiers with neural networks. Ren *et al.* (Ren, He, Girshick, & Sun, 2015) introduced Faster R-CNN, a faster and upgraded version of Fast R-CNN, that replaced the region proposal method with RPN (Region proposal Network), it

also simultaneously predicted object bounds and scores. Joseph Redmon *et al.* (Redmon, Divvala, Girshick, & Farhadi, June 2016) in 2015 introduced a much efficient algorithm for object detection and especially facial recognition called You Only Look Once, a Real time object detection module that is much faster than the RCNN models. YOLO performed Object detection in single stage, thus increasing the inference time. In 2016, Redmon came up with YOLOv2 and in 2018 with YOLOv3 (Joseph Redmon ; Ali Farhadi; 2018) .These versions were bigger, faster and more accurate than its predecessors were. However, as of 2020 YOLO has had many upgrades making it a perfect option for multiple object detection within one scene. S. Saikia et al. (S, E, E, & L, 2017) in their paper have tested a crime scene dataset using the Faster R-CNN and achieved the accuracy of 74.33% and the time taken to detect an object was 0.12 secs.

Samson, Oladipo & Emeka (Francisca, Emeka, & Femi, 2020) have applied YOLO for crime scene evidence analysis. They have trained five classes' objects with 1,173 custom images common to indoor crimes. Further, they deployed it on to android-based forensic case documentation. However, they have only explored the object detection aspect only. Taking their work ahead, this thesis is aimed to make a comprehensive tool for Crime Scene analysis.

1.3 Research gaps identified in the proposed field of the Research

1. Every crime scene is different and has different clues, the foremost gap is training the model for multiple crime scene objects.
2. The next is shooting the video or image of crime scene under adequate lighting condition, the system may perceive the objects differently under different lighting.
3. The accuracy of the object detection and facial recognition may depend on the quality of image and resolution.
4. Detecting objects with similar features can also be a challenge.

1.4 Objective of the proposed study:

The specific objectives of present research are as below:

1. To identify the problems faced by the Investigators of a complex Homicide Crime Scene.
2. To develop a model using Object Detection algorithms to solve these problems.

3. To test the accuracy of the developed model.
4. To run facial recognition and identification of the victim.

1.5 Research Methodology & Detailed research plan

The proposed technology

The proposed model processes the crime scene Images and videos to detect objects and does a facial recognition on the victim to retrieve the records from citizen database. All identified objects are displayed as a list. The model is trained on few different classes of objects such as furniture, weapons, electronics, etc. Every detected object from the image has to belong to a single class of object. The Images for training the model are sourced from open-source platforms. The researcher then trains YOLO to identify objects of these classes and perform a facial recognition as well.

The researcher proposes the use of YOLO (You Look Only Once), as YOLO makes less than half the number of background errors compared to Fast R-CNN. It also learns generalizable representations of objects. (Juan, 2018) YOLO uses features from the entire image to predict each bounding box and also predicts all bounding boxes for an image consecutively. The system divides the input stream into a $S \times S$ grid. If the center of an object falls into a grid cell, that grid cell is responsible for detecting that object. Each grid cell predicts B bounding boxes and confidence scores for those boxes. These confidence scores reflect how confident the model is within the contained box. Based on this score the model predicts its accuracy. YOLO predicts an objectless score for each bounding box using logistic regression. Both image classification and object localization techniques are applied for each grid of the image and each grid is assigned with a label. Then the algorithm checks each grid distinctly and marks the label which has an object in it and also marks its bounding boxes. The labels of the grid deprived of object are marked as zero. By using Bounding boxes for object detection, only one object can be identified by a grid. In case of multiple objects where a bounding box overlaps the other YOLO uses Anchor boxes, anchor boxes have a definite ratio and they try to detect objects that nicely fit into a box with that ratio. YOLO also comes with a complex feature extraction module, it uses a successive 53 convolutional layers of 3×3 and 1×1 layers called Darknet 53. The Yolo is trained with the custom weights to detect and scan a crime scene.

1.6 Chapter Scheme

Chapter 1 Introduction

This is the introductory chapter that gives an overview of the proposed idea. In this chapter the research plan and flow of the thesis is discussed. This chapter iterates the objective, scope and the limitation of the proposed research.

Chapter 2 Review of Literature

Review of literature is the first step to research. This chapter evaluates all the research and work done in the area of forensic photography, object detection and facial recognition. This chapter sets the base for conducting the research.

Chapter 3 Forensics photography

Forensic photography is a technique of photographing crime scene, searches and investigation stages. This keeps a visual record of the crime as well as the investigation. This technique tells a visual story of the crime scene. Since this research is based on images and videos of crime scene, it is very essential for us to explore the forensic photography. Understanding forensic photography and its limitations will help to shape the research better.

Chapter 4 Journey of object Detection

This chapter give a brief walk through in the journey of object detection. Object detection is a very essential aspect of computer vision and imaging. There are many algorithms that enable us to perform object detection and facial recognition. This is very crucial aspect of this research as the proposed research is based on detecting objects in crime scene and facial recognition.

Chapter 5 Experimental Setup and Evaluation

This chapter prepares the model for training and experimentation. The images for the training classes were collected, labelled and augmented. Using the Google Colab the model was trained with object classes and people classes; The images were prepared for training on various models of YOLO. Also, various Evaluation Metrics were discussed.

Chapter 6 Result and Discussion

The model was trained on various models of YOLO, to furnish results on said evaluation Metrics. Results for each class were evaluated and interpreted to compare the different models of YOLO in order to identify the best suitable model for real world application.

Chapter 7 Conclusion and Future Work

This chapter furnishes the results of the proposed research and briefs the research findings. This chapter also highlights the future prospects of the research. This research if implemented can certainly bring about a magnanimous change in the perspective of crime scene analysis. This research aims to multiply the efficiency and accuracy of investigation.

CHAPTER – II

REVIEW OF LITERATURE



In the 21st century, the world is experiencing a significant population explosion, which contributes to significant increase in crime rates. The rise of crime demands a continuous observation and investigation. In such case CCTV and IP cameras has provide necessary surveillance; however, when it comes to large volumes of footages we experience significant challenges. A crime scene investigation demands a manual cataloguing of evidences by an expert investigator. This helps in rightful identification of the victim as well as the evidence.

Historically the images and videos collected from the crime scene is crucial for forensic investigation to retrieve key evidences. The modern world shift to digitisation has helped transform the traditional forensic investigation. The advent of artificial intelligence further has helped revolutionize the digital forensic as well as data collection.

This review of literature explores existing digital forensic investigation and data collection models. This chapter aims to identify gaps in current models and highlight the need for improvement and innovation. It focuses to present current findings to discuss the shortcoming and scope of improvement for the new proposed model. Deep Neural Networks (DNNs) has certainly shows a significant contribution in the improving image classification.

Erhan et al. (2014) while working in object localization in images using DNN-based object mask regression focuses on specific objects such as fire arms and knives.

O'Reilly et al. (2012) in their research use signal processing techniques for concealed weapon detection with the help of neural networks, displaying amazing results with the multimeter wave radar effectively detects concealed weapons.

He, Zhang, and others (2016) concluded that automatic feature representations, unsupervised approaches, multiple instance learning significantly surpass manual feature representation, supervised approaches and supervised learning performance.

Gerga et al. (2016) proposed an algorithm to detect firearms and knives using OpenCV. This proposed algorithm reduces weapon detection efficacy to 35% minimizing false alarms, sacrifice sensitivity as it is vital to not miss even single weapon detection in real world.

Pandey et al. (2016) found that mobile devices, low cost cameras, AI and Machine Learning have enhanced forensic analysis. These have also pioneered research focused on video validation to detect forgery attempts.

Kamenicky et al. (2016) explored and introduced various video analysis methods and tools, addressing the challenge of source of images and videos in criminal investigations.

Amerini et al. (2017) introduced a method to extract composite fingerprint from cell phone cameras using PRNU to identify source from videos shared on social media.

Horsman (2018) proposed a forensic method for detecting and reconstructing cached online video stream data from platforms such as YouTube, Twitter, and Facebook.

Senan (2017) highlighted the issue of over-enhancement in CCTV footage using HE-based methods, which can result in unnatural and washed-out appearances, particularly in low-light conditions with narrow dynamic ranges.

Ayyavoo and Suseela (2018) introduced a color video enhancement technique using Discrete Wavelet Transform and CLAHE, named 'DWT E-CLANE,' which improves facial image recognition.

Hendrawan and Asmiatun (2018) discussed the effectiveness of CLAHE in overcoming the over-enhancement problems associated with HE, noting that various CLAHE variants have demonstrated effectiveness in specific scenarios.

He et al (2014). proposed SPP-net based on Spatial Pyramid pooling that improved the detection and classification time by pooling region features instead of sending each region into the CNN.

Ren *et al* (2015) introduced Faster R-CNN, a faster version of Fast R-CNN, which replaces the previous region proposal method with RPN (Region proposal Network), which simultaneously predicts object bounds and scores.

Girshick et al (2014) proposed a R-CNN algorithm, which was one of the first real target detection model based on convolutional neural networks. The improved R-CNN model gave 66% mAP score making the initial Selective Search to extract approximately 2000 region proposals of each image. Finally, a linear regression model is trained to perform the regression operation of the bounding box for each

extracted image that is processed into SVM classifier. Resulting in improved R-CNN. As a limitation this model had large computation and low efficacy. Also, directly scaling the region proposal to a fixed-length feature vector may cause object distortion.

He, K.M , et al (2015), The ineffectiveness and poor detection issues are resolved by the Spatial Pyramid Pooling (SPP) model. This eliminates the requirement for R-CNN picture blocks with fixed input sizes. The suggested method extracts the features map and only needs to do the convolution computations once. To extract the feature vector of a fixed size, the spatial pyramid pooling layer is added to the final convolutional layer and passed through. In contrast to the R-CNN, Spp-Net only needs to execute feature extraction once, saving on several computations. It still has the same drawbacks as R-CNN, though: 1) Training steps with several steps are complex. 2) More regressors are needed, and separate SVM classifiers must be trained.

Man Ro et al. (2021) introduce a novel method for object classification and localization that employs attentive layer separation. Using ResNet-101 as the backbone network, their approach separates less semantic and semantic layers to handle object classification and localization independently. This technique enhances the effectiveness of object detection by leveraging distinct semantic layers for improved accuracy.

Yu et al. (2021) develop an object detection algorithm that utilizes Region-based Convolutional Neural Networks (RCNN) combined with selective search. Their approach involves extracting around two thousand candidate regions from the initial image, normalizing these regions, and applying Support Vector Machines (SVM) for feature extraction. Non-Maximum Suppression (NMS) is used to select the highest scoring regions, refining the detection process.

Kanimozhi et al. (2021) proposed a lightweight object detection network using MobileNet and Single Shot MultiBox Detector (SSD), termed MobileDet. Their model demonstrated a detection time of 3-5 seconds. In contrast the accuracy decreases when objects are distant than 30 meters from camera. This study is highlights the comparison aspect between detection speed and accuracy at varying distances

Zahisham et al. (2021) resized the images to 224x224 pixels and applied convolutional filters for feature extraction. By fine-tuning a pre-trained ResNet-50 model on various food datasets, including ETHZ-FOOD101, UECFOOD100, and UECFOOD256, they achieve superior performance compared to existing methods, demonstrating fast training and high accuracy.

Deepa et al. (2021) trained their model for real-time tennis ball tracking using YOLO, SSD, and Faster RCNN. The dataset had images captured from multiple angles and lighting conditions. In their research they found SSD providing minimum detecting time with high efficacy.

Garg et al. (2021) Using a 448x448 pixel input image they trained a model for face detection on a YOLO-based architecture. The model predicts bounding box coordinates and class probabilities using Non-Maximum Suppression (NMS). The outcome of the study showed consistent accuracy of 92.2% and improved frames per second (FPS) with lower resolution images, proving YOLO's effectiveness for face detection.

Zhang et al. (2021) using TensorFlow and OpenCV, and training on the WIDER FACE dataset presented a Multi-task Cascaded Convolutional Networks (MTCNN) approach for face detection. As a result, an average precision (mAP) of 85.7% was achieved when compared MTCNN with YOLOv3. MTCNN's performance was superior in detecting multiple faces, especially in complex scenes.

Oumina et al. (2021) uses pre-trained deep learning models such as MobileNetV2, VGG19, and Xception to detect face masks. The method of combining models with classifiers like Support Vector Machine (SVM) and K-Nearest Neighbors (K-NN), they achieve a 97.1% presents high classified performance for face mask detection. They combine these accuracies with a small dataset.

Girshick, (2010) advanced the field with the Deformable Part-Based Model (DPM), extending HOG by using a part-based approach to object detection, which decomposes objects into parts for detection. Girshick later enhanced DPM with mixture models to address real-world variations, influencing subsequent object detectors.

Girshick et al., (2012) The field underwent a major transformation post-2010 with the resurgence of convolutional neural networks (CNNs), which provided robust feature representations and marked a new era in object detection.

Liu et al., (2015) introduced the Single Shot MultiBox Detector (SSD), which improved detection accuracy and speed by employing multi-reference and multi-resolution techniques. SSD detects objects of varying scales on different network layers, significantly enhancing its performance for small objects and achieving a COCO mAP@.5 of 46.5% with a fast version running at 59fps.

Lin et al., (2017) proposed RetinaNet, addressing the accuracy limitations of one-stage detectors by introducing "focal loss," which reshapes the standard cross-entropy loss to focus on hard-to-classify examples, achieving a COCO mAP@.5 of 59.1%.

Law et al., (2018), CornerNet, developed by Law et al. in 2018, shifted the paradigm from anchor boxes to key point prediction, improving performance by predicting object corners and forming bounding boxes from them. This approach resulted in a COCO mAP@.5 of 57.8%.

Zhou et al., (2019) put forth CenterNet achieving a COCO mAP@.5 of 61.1%. It is a key point-based detection model to simplify the detection process. It focuses on object centers and integrates multiple tasks into a single framework.

Carion et al., (2020) introduced DETR, that uses Transformers for object detection. This eliminated the need for anchor boxes and a new level of performance was achieved.

Zhu et al., (2021) proposed Deformable DETR to address DETR's time and performance issues in detection of small objects. This model achieved a COCO mAP@.5 of 71.9%.

Negi et al. (2021) proposed a simplified a Neural network using Keras-Surgeon for efficient face mask detection. The study signifies the efficacy and improved performance of pruning while reducing complexity.

Girshick, (2015) introduced the Fast R-CNN model, which significantly improved upon the original R-CNN by enhancing detection speed and accuracy. On the joint VOC2007 and VOC2012 dataset. The Fast R-CNN achieved a mean Average

Precision (mAP) of 70.0%. The model incorporates three key innovations: (1) it replaces the Support Vector Machine (SVM) used in R-CNN with a softmax function for classification, (2) it introduces the Region of Interest (RoI) pooling layer, derived from the pyramid pooling layer in SPP-Net, to convert candidate box features into a fixed-size feature map suitable for the fully connected layer, and (3) it employs two parallel fully connected layers instead of a single softmax classification layer. Despite these advancements, Fast R-CNN does not fully meet the requirements for real-time detection.

Ren et al., (2016) proposed the Faster R-CNN model, which further advances object detection by replacing the Selective Search method with Region Proposal Networks (RPN) to generate region proposals. The model comprises two main modules: (1) a fully convolutional neural network that generates region proposals, and (2) the Fast R-CNN detection algorithm. These modules share a set of convolutional layers, with the input image passing through the CNN to reach the final shared convolutional layer. This setup allows the network to generate feature maps for both the RPN and the Fast R-CNN detection algorithm. While Faster R-CNN excels in detection accuracy, it still falls short of achieving real-time detection capabilities.

LIDAR, (2023) Object detection has evolved significantly with various technologies and methodologies introduced over the years. Early systems employed LIDAR sensors for vehicle detection and non-intrusive methods like Adaptive Spatial Feature Fusion (ASFF) and Radar Sensors (ASFF, 2023; Radar Sensors, 2023).

Phillips (2021) developed a vehicle distance estimation system using monocular cameras, though accuracy declined with distance. Wang (2021) demonstrated edge detection for vehicle identification, including use in drones. Sokalski (2021) combined edge detection with color identification, while Kanistras (2021) proposed an edge detection method using angle vectors.

Xiao and Kang (2021) emphasized the importance of diverse datasets for training algorithms, and Zoph (2021) highlighted data augmentation strategies to improve accuracy.

Lin (2021) created a YOLO-based traffic counting system, achieving 95% accuracy during the day with improvements for night detection. Tao (2021) optimized YOLO

with a pooling layer and pre-processing for night images, reaching 80.1% accuracy on custom datasets. Corovic (2021) demonstrated YOLO's effectiveness in real-time detection, though occluded objects reduced accuracy. Salarpour (2021) employed Kalman filters and background subtraction for multi-vehicle tracking, achieving 96% accuracy. Phan (2021) introduced an occlusion reduction method with background subtraction, improving detection accuracy to 85% in high traffic. Lu (2021) modified the Region Proposal Network (RPN) to address scale variability, achieving precision scores of 64.1% and 84.8% for different scales.

Redmon et al., (2016) introduced the YOLOv1 object detection model, which marked a significant departure from previous methods by eliminating the need for region proposal extraction. Instead, YOLOv1 utilizes a simple convolutional neural network (CNN) structure. The model processes the entire image as input and directly predicts bounding box locations and categories at the output layer. Specifically, it divides an image into an $S \times S$ grid, where each grid cell predicts B bounding boxes along with their confidence scores. Each cell predicts a total of $B \times (4+1)$ values. YOLOv1 achieves real-time detection with a speed of 45 frames per second (fps) on a Titan X GPU. Despite its fast processing, YOLOv1 has been noted for producing fewer background errors but struggles with recognizing objects in groups.

Redmon et al., (2016) In 2016, Redmon proposed YOLOv2 to enhance both recall and localization while maintaining classification accuracy. YOLOv2 integrates several improvements, including the use of a new feature extraction network, Darknet-19, which consists of 19 convolutional layers and 5 max pooling layers. The model incorporates batch normalization, removes dropout, introduces an anchor box mechanism, and employs k-means clustering on training set bounding boxes. These modifications significantly boost recall and accuracy. However, challenges remain in detecting highly overlapping and small targets.

Redmon & Farhadi, (2018) developed YOLOv3, which is noted for its balanced performance in terms of detection speed and accuracy. YOLOv3 introduces multi-label classification by replacing the original softmax layer with a logistic regression layer for multi-label classification. The model uses a multi-scale prediction approach with upsampling and fusion similar to Feature Pyramid Networks (FPN). YOLOv3

employs a deeper feature extraction network, Darknet-53. Although YOLOv3 improves the detection of small targets and overall speed, it does not significantly enhance detection accuracy, particularly when Intersection over Union (IoU) exceeds 0.5.

Brindha et al. (2021) propose an enhancement to the YOLOv3 algorithm by integrating edge detection for boundary box construction. Their method involves scaling the input image to 416x416 pixels, applying CNN processing, and utilizing edge detection techniques to construct bounding boxes based on threshold values. This approach aims to improve the precision of object detection.

Bhuiyan et al. (2021) apply YOLOv3 for mask detection, using a dataset of 600 images with annotations for mask-wearing and non-mask-wearing individuals. Their method detects bounding box coordinates and determines mask presence, contributing to improved face mask detection accuracy.

Liu and Zhang (2021) improved the YOLOv3 model for traffic conditions, achieving a Mean Average Precision (mAP) of 91.12% with their F-YOLOv3 algorithm, surpassing Faster R-CNN and YOLOv3. Redmon (2018) integrated classification and localization into a single convolutional neural network, though it struggled with smaller objects and varying aspect ratios.

Chandan (2021) used OpenCV and SSD for vehicle detection, providing a benchmark against YOLO models. Chen (2021) compared YOLOv3 and SSD, finding YOLOv3 more effective for high-resolution traffic detection, while Kim (2021) reported YOLOv4 achieving 98.1% precision compared to SSD and Faster R-CNN.

(Liu et al., 2016) proposed the SSD (Single Shot MultiBox Detector) model, which combines the regression idea from YOLO with the anchor box concept from Faster R-CNN. SSD improves multi-scale object detection by using both lower and higher-level feature maps. Its base architecture is VGG, with the last two fully connected layers replaced by convolutional layers. SSD incorporates the anchor mechanism from Region Proposal Networks (RPN). It achieves a mean Average Precision (mAP) of 74.3% on VOC2007 at 59 fps on a Nvidia Titan X. However, SSD's performance diminishes for small targets, and it struggles with redundant detections due to independent feature maps at different scales.

Ahmed et al. (2021) use SSD with MobileNet for pedestrian detection, focusing on real-time accuracy with the COCO dataset. Their model excels in detecting overlapping pedestrians, providing a balance between speed and accuracy.

Bochkovskiy, (2020) In 2020, Bochkovskiy introduced YOLOv4, which set new benchmarks for the balance of speed and accuracy (Bochkovskiy, 2020). YOLOv4 builds on the original YOLO framework by incorporating several innovations, including Weighted Residual Connections, Cross Stage Partial connections, Cross Mini-Batch Normalization, Self-Adversarial Training, Mish activation, Mosaic data augmentation, DropBlock, and CIoU loss. The model uses CSPDarknet53 as its backbone network and includes an SPP module to expand the receptive field for better feature separation. YOLOv4 replaces FPN with PANet for path aggregation and retains the head structure from YOLOv3. Compared to YOLOv3, YOLOv4 improves accuracy and speed by 10% and 20%, respectively.

Bhambani et al. (2021) propose a YOLOv4-based model that classifies and detects objects in three categories: people, masked faces, and unmasked faces. With components like CSPDarknet53 and SPP, and calibration for social distancing, their model achieves a mean average precision (mAP) of 95% and a frame rate of 38 FPS on NVIDIA Tesla P100 GPU. The YOLOv4 model has shown high accuracy with a mean average precision of 98.1%. In contrast the YOLOv5 has shown further improvements (YOLOv5, 2023). This research enhanced an object detection capability in South Asia using a robust system to process and execute a diverse dataset of 21 vehicle classes.

Li, Lin, Shen, Brandt, and Hua (2015) introduced a CNN cascade for face detection. Their proposed model balances high discriminative capability and efficiency, by operating at multiple resolutions while only evaluating high-resolution candidates only. A CNN-based calibration stage is integrated after each detection stage to enhance efficacy and lowers number of subsequent stages.

RoyChowdhury, Lin, Maji, and Learned-Miller (2015) proposed a Bilinear CNN (B-CNN) method that bridges texture models and part-based CNN models. The B-CNN consists of two CNNs whose convolutional-layer outputs are multiplied using an outer product to create a bilinear feature descriptor. This approach models part-based

representations and resembles Fisher vectors, integrating local features with cluster center membership through outer products.

The B-CNN architecture is trained by back-propagating gradients from a task-specific loss function, starting with pre-trained networks (e.g., AlexNet) and fine-tuning on face images. The bilinear layer, similar to a quadratic polynomial kernel in SVMs, improves recognition performance. The B-CNN showed substantial performance gains on face recognition benchmarks with pre-trained networks.

Chaudhari et al. (2021) proposed VGG-16 for identifying tree species from WorldView-3 satellite images. They analysed their approach with Random Forest and Gradient Boosting classifiers, using data from eight visible infrared bands to enhance the detection of the tree species.

Liu (2021) enhanced Faster R-CNN by integrating hard negative sample strategies and feature sharing training. This method retraines the model with negative samples and improves detection speed and precision. The study analyses YOLO, SSD, and RFCN, signifying the strengths of each in terms of detection speed and precision.

Rezaee et al. (2021) use WorldView-3 satellite imagery and VGG-16 for detecting individual tree species. Their pipeline leverages eight visible infrared bands and compares results with Random Forest and Gradient Boosting classifiers, showcasing advancements in forest monitoring.

Han, C., Liu et al., (2018) suggested conducting research on a cutting-edge method to determine the clustered image's shape. The main concept in the image's metamorphosis is the shift from a cluttered image to a clear shape. The pairs between the local shape picture and the requested shape are identified by the point-based descriptive PAD (Pyramid of arc length descriptor). Wavelet wave transforms and Fourier transforms were used to convert the shape into domains in order to measure it. Afterwards, a number of shape descriptors were put forth to gauge the degree of shape similarity. Triangle regions are used as a set of reference points to describe the shapes. Nonetheless, the current methods and shape descriptors for matching shapes are made for matching shapes.

He, K., Zhang et al. (2016) proposed a research on the classical and deep learning methods in object detection. The model's approach is centered on operation and real-

time performance, and precise detection. The difficulties in object detection using deep learning methods. Three processes make up the subtraction method: object identification, background updating, and background modeling. The backdrop subtraction approach updates in real time and works similarly to the frame difference method. The deep learning approach-based object detection model. The models later put out a novel concept.

Numerous researchers have addressed the benefits and drawbacks of the literature reviews that have been covered, but there are still certain problems with object detection and facial recognition. In order to overcome the problem of uncontrolled environmental problem, face direction recognition where the more Headshot and real-time captures help to verify the effectiveness of the Object detection model, enabling higher accuracy.

CHAPTER – III

FORENSIC PHOTOGRAPHY



The law enforcement community must continually assess its mission to ensure the effective use of photography. This ongoing review requires exploring various potential applications, as photographic responsibilities and objectives can vary significantly depending on the situation.

A critical application of photography in law enforcement is the documentation of crime scenes. A comprehensive visual record of the scene is essential for thorough investigations and subsequent prosecutions. Thus, it is crucial to address theoretical, legal, and technical considerations before conducting on-site photography. It is important to note that poorly planned, executed, or displayed photographs can adversely impact the success of the crime scene investigation. Hence, crime scene photography is a fundamental component of the entire investigative process.

Before beginning systematic photography, the purpose and fundamental principles must be established. The primary goal of crime scene photography is to create a detailed visual record of the scene and its significant elements. Photography should capture a logical sequence that illustrates the "story" of the scene. It is essential to keep the scene undisturbed to ensure that the photographs accurately represent the original conditions. The comprehensive documentation of the scene should not be compromised by concerns about the cost of film; thoroughness is paramount. When in doubt, it is better to take additional photographs rather than risk missing important details that may become significant later.

Theoretical considerations in crime scene photography involve creating a sequence of images that cover all relevant aspects of the scene. This approach generally follows a "general to specific" progression, involving three main types of shots: 1) long-range, 2) mid-range, and 3) close-up. Long-range photographs might include aerial views or wide shots of a scene, such as a hallway, while mid-range shots provide a more focused view from a distance of ten to twenty feet. Close-up photographs, taken from five feet or less, focus on detailed evidence not visible in wider shots. Each stage of the crime should be separately documented, illustrating the sequence of events from the approach to the scene, the commission of the crime, and departure.

The perspective of the camera is essential to making sure that photos accurately capture the scene. In a living room, for example, long-range images should show the

area from eye level, mid-range shots should have enough detail to connect the various aspects of the scene, and close-ups should highlight certain pieces of evidence.

Whenever possible, measurement scales should be utilized to appropriately illustrate relationships between size and distance. To prevent clutter, nevertheless, it can also be necessary to shoot pictures without these scales. Usually, it helps to jot down the position of each shot so that you have a reference point for figuring out the perspective of the pictures. For better understanding, this sketch can be marked up and affixed to the pictures.

In general, there are five sorts of photographs: 1) images of the site, which display the different sections of the crime scene; 2) shots of the surroundings, which help determine the kind of crime; and 3) photos of the results, which depict the course and consequences of the crime, 4) tangible evidence images, which record evidence in connection with the incident.

Contrary to the belief that analog photos were inherently trustworthy, they were simply harder to forge due to the complexity of the process. For instance, Nicéphore Niépce created the first surviving photograph in 1825, but even early photography faced issues with forgeries. By the 1860s-1870s, expert photographers were already creating convincing forgeries, such as the altered image of General Grant during the American Civil War, where only his face was genuine.

Today, digital imaging allows for highly convincing fakes even with low-power devices like smartphones. Forged images can misrepresent various contexts, from fashion to military displays, raising serious concerns when used in social, political, or military contexts. For example, falsified images of missiles could mislead military decisions, while manipulated academic or medical images can affect research and insurance claims.

Identifying the authenticity of an image by visual inspection alone is unreliable, as genuine photos can appear fake and vice versa. This highlights the need for sophisticated Digital Image Forensics. This field aims to trace an image's history and verify its authenticity by analyzing subtle traces of manipulation. Digital Image Forensics operates without access to the original, unaltered image and relies on detecting these subtle traces of processing to uncover manipulations. Given the

complexity of image processing, tracing the full history of an image can be challenging, as each edit diminishes evidence of prior modifications.

3.1 The history of forensic photography

The history of forensic imaging traces back to the invention of the camera obscura, the earliest pinhole camera. These early pinhole cameras were employed by scientists to observe the sun and by artists for sketching.

Discrepancies in historical dates often arise due to multiple associated milestones: the initiation of research, completion of results, patenting, and public announcement. For instance, the evolution of the camera obscura involved several key developments. In 1550, Girolamo Cardano introduced a lens to the camera obscura design, a term he coined based on its resemblance to lentils. Giovanni Battista della Porta enhanced the design in 1558 by incorporating lenses and curved mirrors to produce upright images, though this was not published until 1588. The addition of a diaphragm, attributed to Daniele Barbaro in 1568, completed the foundational elements of early photographic cameras.

Progress in photographic technology continued slowly. In 1614, Angelo Sala noted that sunlight darkened silver nitrate, though its significance was not understood at the time. Later, in 1725, Johann Heinrich Schulze demonstrated that light darkens certain silver salts. In 1737, Jean Hellot used a photographic process to reveal secret writings by exposure to light, possibly coining the term "photography," meaning "writing with light"

Carl Wilhelm Scheele's 1777 discovery that silver chloride turns black with light and can be dissolved by ammonia did not lead to practical photographic applications. The first documented photographic attempt using a camera obscura was made by Thomas Wedgwood in 1795, which failed due to underexposure and difficulties in fixing the image.

In 1800, Sir William Herschel's discovery of the invisible infrared spectrum through a simple experiment significantly impacted law enforcement photography. He used a beam splitter to separate white light into its component colors and discovered the infrared region of the electromagnetic spectrum (Scott, 1969, Vol. 2, p. 79).

Joseph Niépce's photography experiments began in 1816, with John Herschel discovering the use of hydrosulfite of soda to dissolve silver salts in 1819. This innovation marked him as a pioneer in photography.

In 1835, William Henry Fox Talbot produced the first photographic negative, followed by Sir John Frederick William Herschel's discovery of hyposulphite of soda for fixing photographic images (Hedgecoe, 1980, p. 22; Davis, 1995, pp. 6–7). Talbot's calotype process, patented in 1841, involved a silver chloride-coated light-sensitive paper and was also commercially successful (Davis, 1995, pp. 5–6; Hedgecoe, 1980, p. 22).

The 1850s saw further innovations: Aimé Laussedat developed photogrammetry, Frederick Scott Archer created the collodion wet plate process, and Sir George G. Stokes discovered UV fluorescence and formulated Stokes' Law, foundational for fluorescent photography in law enforcement.

In 1854, new processes like the ambrotype and carte-de-visite emerged, offering cheaper alternatives to the daguerreotype. The ambrotype, invented by J.A. Cutting, and the carte-de-visite, developed by André Adolphe-Eugene Disdéri, were easier and more affordable (Davis, 1995, p. 10; Spira, 2005, et al., pp. 55–62). The tintype, invented by Hamilton Smith in 1855, was another economical option.

The evolution of photographic technology continued with significant milestones, including Gaspar Felix Tournachon's aerial photograph of Petit-Becetre in 1858, captured from a hot-air balloon using the wet plate process (Jeffery, 1998, p. 220).

- **Early Use of Photography in Forensics:** Forensic image analysis has roots as early as 1851 with the examination of a faked color daguerreotype. The pivotal moment came in 1859 when the U.S. Supreme Court ruled on the admissibility of photographs as evidence, setting a precedent for the use of photographic evidence in legal proceedings.
- **Stereo Photography and VR Integration:** While stereo photography was popular in the mid-19th century, it was not widely adopted by law enforcement at the time. However, modern forensic techniques now integrate similar principles with VR and photogrammetry for detailed crime scene analysis.

- **Color Photography Advances:** In 1861, Maxwell and Sutton's successful color separation negatives laid the groundwork for modern color photography. This development, despite being achieved with orthochromatic film, demonstrated the potential for accurate color reproduction.
- **Civil War Photography:** Mathew Brady and other photographers documented the American Civil War, influencing public perception and raising early concerns about photographic accuracy and representation.
- **Crime Scene Photography:** By 1867, crime scene photography had begun, and early marketing for crime scene cameras mirrored modern sales tactics for digital photography systems.
- **Spirit Photography Fraud:** The 1860s and 1870s saw the rise of fraudulent spirit photographs, which were faked using double exposure techniques. This early form of photographic deception underscores the need for forensic analysis to verify image authenticity.
- **Technological Advances:** Key developments included Hermann Wilhelm Vogel's dye-sensitizing technology in 1873, which extended the color sensitivity of black-and-white films, and the rise of flexible transparent film patents in the late 19th century.
- **Court Admissibility:** Throughout the late 19th and early 20th centuries, various court cases established standards for photographic evidence, including the admissibility of photographs, X-rays, and color images.

Aerial Photography: The late 19th and early 20th centuries also saw advancements in aerial photography, starting with Alfred Nobel's rocket-mounted camera in 1897 and the Bavarian Pigeon Corps' use of pigeons for aerial shots.

Modern Developments: By the 1930s, flash bulbs and new color film technologies improved forensic photography. The development of stroboscopic flash systems, dye-destruction color film, and instant photography by Edwin Land in the mid-20th century further advanced forensic imaging capabilities.

FBI Laboratory Developments: In 1942, the FBI Laboratory separated its photographic operations into processing and special photographic units, which eventually became the Forensic Audio, Video, and Image Analysis Unit.

State-Level Crime Laboratories: Crime laboratories across states have evolved differently. Some are part of state police, while others, like Wisconsin's, fall under the Department of Justice. Wisconsin's crime lab, established in 1947, recognized photography as a distinct forensic discipline, which helped forensic photographers gain equal classification and pay as other forensic scientists.

Technological Impacts:

- **1940s:** Introduction of color photography for mug shots.
- **1980s:** Use of Polaroid print film for instant booking photographs, and later, digital photographs which almost eliminated identity-swapping issues.
- **1957:** Introduction of the videotape recorder, which replaced motion picture film for video recording.
- **1963:** Launch of Polaroid Polacolor instant print film.
- **1965:** Introduction of Super 8mm movie equipment and fully automatic electronic flash units.

Legal Developments:

- **1884:** Barrow-Giles Lithographic Co. v. Sarony established that photographs are documents under the U.S. Constitution.
- **1948:** Tennessee Supreme Court ruled that the Best Evidence Rule does not always apply to photographic evidence.
- **1951:** Requirement for photographs to be submitted to the opposing party before being admitted as evidence.
- **1970s-1980s:** Important court cases affirmed the admissibility of photographic and video evidence, and the development of the ACE-V methodology for latent print analysis also influenced forensic photography.

ACE-V Methodology: Developed in the 1970s, ACE-V stands for Analysis, Comparison, Evaluation, and Verification, and is used in both latent print and photographic comparisons.

Forensic Imaging Organizations:

- **1973:** Formation of the American Society of Crime Laboratory Directors (ASCLD).
- **1989:** Founding of the Law Enforcement and Emergency Services Video Association (LEVA).
- **2002:** Digital and multimedia disciplines, including image and video analysis, were formally recognized.

Technological and Software Advancements:

- **1970s:** Introduction of the VHS format and laser dye-staining for latent prints.
- **1980s:** Availability of 35mm point-and-shoot cameras and personal computers capable of digital image processing.

Significant Court Case (1987): U.S. v. Alexander highlighted the importance of expert testimony in photographic comparisons, particularly in forensic contexts.

The evolution of forensic imaging from the late 20th century into the early 21st century highlights significant developments in technology, professional standards, and legal considerations:

1. Formation of SWGIT:

- In 1989, a symposium in Las Vegas led to the creation of a new group focused on imaging technologies in forensic science.
- This group evolved into the Scientific Working Group on Imaging Technologies (SWGIT) in 1998, with a video subcommittee added in 2001.
- SWGIT aims to integrate imaging technologies within the criminal justice system, providing guidelines for the capture, storage, processing, analysis, transmission, and output of images. It includes representatives from law enforcement, academic institutions, and corporations.

2. Impact of Professional Organizations:

- **1997 IAI Resolution 97-9:** Recognized digital imaging as a valid technology in forensic science, contingent on equipment specifications, quality control, and the expertise of the imaging specialist. This resolution came between the Frye and Daubert standards, addressing issues like general acceptance and methodology.
- **1999 Formation of CFSO:** Aimed to inform Congress about the need for funding for forensic science disciplines beyond DNA, including showcasing various forensic disciplines to lawmakers.

3. Technological Advancements:

- **1992-1995:** Development of crime scene sketching programs like Fire Zone CAD and Crime Zone CAD, which could link to laser crime scene mapping and automated panoramic cameras. These tools enhanced crime scene documentation and visualization.
- **1999:** Introduction of automated panoramic cameras capable of creating QuickTime files linked to crime scene sketches with photogrammetry capabilities.

4. Court Cases and Legal Developments:

- **1987:** U.S. v. Alexander underscored the significance of expert testimony in photographic comparisons.
- **1990s-2000s:** Continued reinforcement of the general foundations for admitting videotapes and photographs into evidence, focusing on relevance, accuracy, and the probative versus prejudicial balance.
- **2005 Wisconsin State Attorney General's Office Memo:** Confirmed no legal requirement for a chain of custody for digital photographs and videos, though a chain of custody is required for physical evidence photographs and videos.
- **1999 Florida Case (Dolan v. State):** Addressed chain of custody issues related to digital photography and video evidence.

- **2007 IAI Resolutions:**

- **Resolution 2007-8:** Rejected the use of optical watermarks for authenticating digital images.
- **Resolution 2007-7:** Recognized the validity of photographic comparisons and outlined limitations.

Photogrammetry**Crime Scene Documentation**

Adequate documentation of a crime scene involves three main activities:

1. **Photography:** Capturing visual evidence and context.
2. **Measurement:** Measuring the crime scene and evidence within it.
3. **Sketches and Diagrams:** Creating detailed diagrams of the scene.

Challenges and Potential Solutions

- **Eliminating Measurement:** The suggestion of eliminating traditional measurement methods is not proposed. Instead, photogrammetry is presented as a supplement to traditional techniques. While photogrammetry may not entirely replace traditional measurements, it offers a means to enhance accuracy and efficiency.

Photogrammetry Overview

- **Definition:** Photogrammetry involves:
 - Photographing an object.
 - Measuring the object's image on the photograph.
 - Reducing measurements to a usable form, such as a map or diagram (Moffett & Mikhail, 1980; Slama, 1980).
- **Applications:** Photogrammetry is used in various fields, including mapping the moon and crime scene documentation. It integrates photography with measurement, providing an alternative or supplement to traditional methods.

Benefits of Photogrammetry

1. **Complex Scenes:** Useful in scenes with numerous or intricate pieces of evidence where traditional measurements may be impractical (Baker & Fricke, 1986).

2. **Uncertainty:** Helps in situations where the importance of evidence may not be immediately clear, aiding in later reconstruction (Whitnall & Millen-Playter, 1988).
3. **Adverse Conditions:** Effective in unfavourable weather conditions where traditional measuring methods may be difficult or impossible (Baker, 1983).
4. **Limited Resources:** Provides a solution when the investigator is working alone or lacks the usual equipment (Baker & Fricke, 1986).
5. **Urgent Situations:** Useful in time-constrained situations, such as high-traffic areas or urgent calls.
6. **Minor Scenes:** Helps in documenting scenes that may initially seem minor but could later turn out to be significant.
7. **Insurance Shots:** Allows for additional documentation that might be used later to extract more information from initial photographs.

Comparison to Advanced Systems

- **Total Stations:** Advanced systems like Total Stations, which use laser sighting and computer diagramming to create 3D models, are highly effective but not universally accessible. Photogrammetry provides a more accessible alternative, filling the gap between hand-drawn diagrams and advanced measurement systems.

Digital Image Forensics in Practice

Digital Image Forensics draws from Digital Steganography and Digital Watermarking, which both conceal information in images to protect ownership and verify integrity.

- **Steganography** hides messages within images in a way that is undetectable to the human eye.
- **Digital Watermarking** embeds a visible or invisible signature into images to indicate ownership or ensure integrity. Watermarks can be robust (surviving processing) or fragile (damaged by alterations).

Unlike these methods, Image Forensics does not require the original image or additional information about it. It uses a "blind" approach to analyse images for authenticity, without needing specific hardware.

1. Image Generation Process

Using processing traces, forensic tools examine an image's past. Important phases consist of:

- **Acquisition:** The process of obtaining light, which leaves recognizable traces, through lenses and sensors. Certain brands and devices have distinct acquisition footprints that include things like Color Filter Array (CFA) patterns, lens aberrations, and sensor noise (Photo Response Non-Uniformity, PRNU).
- **Coding:** Certain artifacts, like quantization and blocking artifacts, are left behind by JPEG compression. These may reveal forgeries by assisting in the identification of compression parameters and determining whether a picture has been recompressed.
- **Editing:** Tools can be used to alter images for either benign or malevolent ends. Resampling (rotation, scaling) is one type of editing that might generate periodic artifacts. Changes to the image's look can occasionally mask earlier manipulation.

Key Points:

- **Acquisition Footprints:** Indicate the type, brand, and model of the capturing device, and can reveal inconsistencies if splicing has occurred.
- **Coding Footprints:** Show compression parameters and can identify multiple compression instances.
- **Editing Footprints:** Include modifications from editing tools, which can obscure tampering or alter the image's message.

Contrast enhancement, commonly achieved through histogram equalization, adjusts pixel intensity values to improve image quality but can introduce artefacts detectable in the histogram, such as unexpected peaks and gaps. Median filtering, used for denoising and smoothing, may also obscure previous processing traces; it is detectable through increased pixel value similarity and block-wise correlations. JPEG re-

compression can reduce visibility of manipulation traces by smoothing out artefacts but can be identified by anomalies in grid alignment or quantization tables. Tampering detection involves various techniques: cut & paste forgery can be detected by grid misalignment in JPEGs, inconsistencies in device artefacts, and resampling traces; copy-move forgery detection uses block-wise analysis or robust local descriptors like SIFT or SURF to identify duplicated regions; seam carving, a content-aware resizing technique, introduces specific traces that can be detected through seam classifications; and digital inpainting, which reconstructs or removes parts of images, can be identified by detecting patterns consistent with inpainting methods. Each manipulation technique leaves unique traces or inconsistencies that forensic algorithms analyse to detect and understand image alterations.

Digital Image Counter-forensics

Until recently, counter-forensics—strategies developed to evade forensic analysis—received minimal attention. Adversaries, who possess knowledge of signal processing similar to forensic analysts, aim to manipulate images while making their alterations undetectable. This discipline, known as counter-forensics (or anti-forensics), includes techniques designed to hide, remove, or falsify traces of illicit processing. Counter-forensic techniques often leave their own detectable traces, which forensic analysts can exploit to identify and address limitations in current forensic tools. Presently, counter-forensics can obscure traces of JPEG compression, resampling, filtering, and histogram manipulations. The ongoing interplay between forensic and counter-forensic methods highlights the need for improved forensic tools and approaches.

The Image Dependency Problem

Current image forensics tools focus primarily on analysing single images, but understanding relationships between groups of images—image dependencies—can be equally or more important. Image dependencies can reveal how images relate to one another, identify clusters of images from the same source, and track how image usage evolves over time and across different contexts. For example, analysing dependencies can expose how certain iconic images, like those of the polar bear, the Afghan girl, or significant historical events, have been duplicated and manipulated online. Similarly, images of famous paintings, such as the Mona Lisa, often vary in color, size, and detail despite originating from the same artwork.

Formalizing and comprehending these linkages is a challenge in the study of picture dependencies since many images have actual origins and transformations that are not always recognized or known. In order to solve this, we need to blend human thinking with automated data collecting to deduce logical linkages. In order to make relationship analysis practicable, the chapter will discuss state-of-the-art methods for analysing sets of related photos, explain the idea of detecting dependencies, and present certain presumptions regarding typical image duplication procedures.

Pointers to Near-Duplicates Analysis

The literature on image retrieval makes a distinction between two kinds of image duplication: Near Duplicate (IND) detection, which locates variants of pictures that have been altered through different processing approaches, and Exact Duplicate (IED) detection, which discovers exact copies of a reference image. Scene (such as backdrop alterations, occlusions), Camera (such as perspective shifts, zoom), Photometric (such as lighting, exposure adjustments), and Digitization (such as compression, recoloring, scaling, and cropping) are the categories into which modifications that result in near-duplicates can be divided.

Efficient data handling is necessary because IED and IND detection methods frequently require huge image datasets and strict time limits. Robust descriptors such as Scale Invariant Feature Transform (SIFT) (Lowe, 2004), which has been applied in several research (Ke et al., 2004; Foo et al., 2007a; Zhu et al., 2008), are commonly used to represent images. Adding several descriptions together can increase accuracy.

While these methods can cluster similar images, they often fail to reveal relationships between images within each cluster. A notable attempt to address this is image archaeology (Kennedy and Chang, 2008), which uses binary detectors to analyze near-duplicate connections, producing a Visual Migration Map. Despite its promise, this system lacks a rigorous theoretical framework, does not account for exact duplicates, and does not provide a confidence score or parameter estimation for relationships.

Although similar photos can be clustered using these methods, links between images within each cluster are frequently not revealed. Image archaeology (Kennedy and Chang, 2008) is a noteworthy effort to solve issue, as it creates a Visual Migration

Map by analysing near-duplicate connections using binary detectors. Though promising, this system lacks a sound theoretical foundation, does not take precise duplication into account, and does not offer relationship parameter estimation or a confidence score.

De Rosa et al. (2010) presented a methodology that uses pairwise comparison of near-duplicates to formally formalize picture associations. Their method separates a picture into "noise," or content, and uses these two components as a fingerprint to quantify relationships between them. This technique handles compressed images, handles geometric transformations more broadly, and handles exact duplicates more skilfully.

Dias et al. (2012) introduced an analogous technique called image phylogeny, which uses dissimilarity measures to examine near or exact duplicates. In contrast to De Rosa et al., Dias et al. build their dependency graph, the Image Phylogeny Tree, using minimal spanning trees as opposed to heuristic criteria, and process the complete image instead of only the noise component.

Decision Fusion in Digital Image Forensics

Image forensic research has predominantly concentrated on detecting artefacts introduced by single processing tools. However, in tamper detection, the specific artefacts to be identified are often unknown in advance. This necessitates the application of multiple tools designed for different scenarios. Two primary challenges arise: (i) creating an effective strategy to consolidate the information from various tools into a unified output, and (ii) managing the uncertainty caused by error-prone tools. This process of integrating multiple data sources to form a consistent and useful representation is known as information fusion.

In this thesis, a solution to these challenges is proposed through a fusion framework based on Fuzzy Theory. Fuzzy systems are beneficial in applications where reasoning must be resilient to noise, approximation, or imprecise inputs. A practical implementation of this framework is described, including experiments that test its effectiveness in a realistic scenario. In these experiments, five forensic tools utilized JPEG artefacts to detect cut-and-paste tampering within specific regions of an image. The results demonstrate the framework's effectiveness, particularly in comparison to traditional methods.

Information Fusion in Image Forensics

Information fusion involves integrating multiple data sources and knowledge representations to produce a coherent and accurate representation of the real-world object. In image forensics, each technique is tailored to detect specific footprints left by different processing tools. However, forensic techniques are not infallible; they are subject to uncertainties and inaccuracies due to various factors, including tool settings, image characteristics, and deviations from the tool's working assumptions.

Typically, multiple processing tools are used to create altered images rather than relying on just one. As a result, using a single detection method may not be sufficient. Instead, a range of tools is applied, each producing a different type of output, such as probabilities, scalar values, or binary results. This variety of outputs complicates the process of making a unified judgment about an image's authenticity. Simple methods, like binary OR majority voting, may not always yield satisfactory results due to their limitations.

Although machine learning approaches like Support Vector Machines (SVM) and Neural Networks (NN) offer more complex solutions, they come with challenges, including increased processing demands and the need for retraining when new tools are introduced.

To address these issues, the chapter introduces a fusion framework based on Fuzzy Theory. This approach aims to effectively manage and integrate the uncertain and varied outputs from different forensic tools into a single, final decision.

General Pointers to Information Fusion

Information fusion is the process of combining facts and data about a single real-world entity from various sources to produce a meaningful, accurate, and cogent representation. It is also known as decision combination, expert conciliation, knowledge integration, and decision fusion. This method is widely applied in many different domains, including imaging, biometry, satellite imaging, remote sensing, and speech and speaker recognition. For example, merging information from speech recognition, iris scans, gait analysis, and fingerprints improves the accuracy of biometric verification.

Information fusion originated from the goal of combining the outputs of different classifiers to increase classification accuracy.

The approaches that are pertinent to Image Forensics are included in the classification

Categories of Information Fusion

1. Feature Level Fusion

- **Concept:** Combines features extracted from different tools before classification.
- **Advantages:** Can achieve higher discriminative power by integrating diverse features.
- **Challenges:** Issues may arise with conflicting, redundant, or high-dimensional features, necessitating complex feature selection processes.

2. Measurement Level Fusion

- **Concept:** Combines scores computed independently by each tool based on its own features.
- **Advantages:** Simpler and less data-intensive compared to feature level fusion. **Challenges:** Sensitive to noise and uncertainty; requires consistency in score representation. Methods include:
 - **Classification Techniques:** Use classifiers like SVMs, neural networks, and decision trees to process aggregated scores.
 - **Combination Techniques:** Aggregate scores using methods like linear combinations, min, max, mean, median, product rules, Bayesian models, and non-traditional approaches like Fuzzy Theory and Dempster-Shafer Theory.

3. Abstract Level Fusion

- **Concept:** Combines class labels assigned by each classifier to make a final decision.
- **Advantages:** Universal applicability as all classifiers can provide labels.

- **Challenges:** Limited information available; decisions are often made using majority voting, weighted voting, or AND/OR rules.

3.2 Information Fusion in Image Forensics

In image forensics, tasks like source identification or forgery detection can be approached as classification problems, enabling fusion at feature, measurement, or abstract levels.

1. Fusion at Feature Level

- **Examples:**
 - **Camera Model Identification:** Fusion of similarity measures, image quality metrics, and Wavelet coefficients to identify camera models.
 - **Scanner Model Identification:** Uses noise statistics features and gray-level co-occurrences to differentiate scanner brands.
 - **Device Class Identification:** Combines noise statistics and color interpolation coefficients for device classification.
 - **Forgery Detection:** Combines features from multiple detectors (e.g., copy-move forgery detectors) to enhance detection accuracy, as demonstrated by (Chetty and Singh, 2010).

2. Fusion at Measurement Level

- **Examples:**
 - **Fuzzy-Based Framework:** Introduced by (Barni and Costanzo, 2012b) for combining scores from heterogeneous tools.
 - **Dempster-Shafer Theory:** Proposed by (Fontani et al., 2013) for combining evidence in a flexible manner without relying on a priori probabilities.

3. Fusion at Abstract Level

- **Examples:**
 - **Weighted Majority Voting:** Combines outputs of multiple tools using various voting mechanisms.
 - **Behavior Knowledge Space and Naive Bayes:** Used in conjunction with weighted majority voting to improve detection performance.

3.3 Foundations of Fuzzy Theory

Fuzzy Theory is integral to the proposed fusion framework, as it handles imprecise, noisy, and uncertain information effectively. It provides a way to reason about data in a more flexible manner compared to traditional binary logic, allowing for a more nuanced integration of data from diverse sources. The principles of Fuzzy Theory will be further explored and applied in subsequent chapters of the thesis.

Digital Image Counter-Forensics

Digital Image Counter-forensics involves developing techniques to mislead forensic analysis by concealing, removing, or falsifying traces that forensic tools detect. Despite the field's progress, challenges remain, particularly with robust forensic methods like SIFT (Scale Invariant Feature Transform). Recent research has introduced methods to remove SIFT key points to bypass detectors, alongside algorithms for detecting such removals and injecting fake key points to mislead detection efforts. The formalization of forensic and counter-forensic problems involves understanding the image generation process, distinguishing between original, processed, authentic, and manipulated images, and modeling forensic analysis as a classification problem. Counter-forensic attacks can be integrated, altering the generation process to prevent detectable traces, or post-processing, modifying images after generation. Attacks may be targeted, designed to counter specific forensic methods, or universal, aiming to evade detection by various tools. The field continues to evolve, balancing effective counter-forensic techniques with maintaining image quality.

In the realm of JPEG compression, specific artifacts are created, including the comb-like pattern in the Discrete Cosine Transform (DCT) coefficient histogram and blocking artifacts in the spatial domain. Initial counter-forensic techniques aimed to obscure these footprints. Stamm et al. (2010a) introduced anti-forensic dithering to mitigate gaps in the DCT histogram by adding noise that mimics the unquantized coefficient distribution. Lai and Böhme (2011) identified peculiar traces left by anti-forensic dithering, developing detectors to reveal these traces and subsequently refining their dither technique to counteract these detectors. Valenzise et al. (2011b) evaluated the perceptual impact of dithering and introduced a detector based on total variation (TV) to identify dithered images, though this approach was not robust

against techniques like those by Fan et al. (2013b), which use TV minimization to remove JPEG blocking artifacts and include a de-calibration stage. Li et al. (2012) examined how random DCT modifications disrupt coefficient correlations and proposed methods to detect these changes, outperforming previous techniques. Fan et al. (2013a) challenged assumptions about the Laplacian distribution of DCT coefficients and introduced a non-parametric smoothing technique to counteract anti-forensic dither.

Counter-forensic methods also target detection of multiple JPEG compressions, which can indicate image manipulation. Chunhui et al. (2012) proposed removing double quantization artifacts, while Milani et al. (2013) altered the first digit probability mass function to align with a single compressed image, countering detectors based on Benford's law.

To address histogram manipulations, contrast enhancement can create impulsive peaks and gaps in the grayscale histogram. Cao et al. (2010a) employed Gaussian dithering during contrast remapping to remove these features, impairing known detectors. Barni et al. (2012) introduced a universal post-processing technique to modify manipulated image histograms to match authentic images from a database, effectively neutralizing contrast enhancement detectors. Lin et al. (2013) developed a contrast enhancement detector for color images, addressing alterations in high-frequency components.

Resampling evidence is crucial for detecting cut & paste forgeries, as it often requires resizing or rotating parts of an image. Kirchner and Böhme (2007; 2008) proposed a method to avoid periodic dependencies by adding Gaussian noise to high-frequency pixels during resampling and applying median filtering to low-frequency pixels. Fontani and Barni (2012) focused on detecting median filtering traces by optimizing sliding window operators to remove filtering footprints.

Forging the source of an image, such as altering PRNU or CFA patterns, can obscure the origin of a digital image. Gloe et al. (2007b) and Kirchner and Böhme (2009) proposed methods to replace authentic PRNU and CFA patterns with target patterns, respectively. Rao et al. (2013) challenged source identification methods and the Triangle Test, providing new insights into source forgery techniques.

Finally, counter-forensics has been framed as a game theory problem to better understand interactions between forensic analysis and counter-forensic strategies. Stamm et al. (2012) and Barni (2012) applied game theory to evaluate adversarial strategies and optimal forensic countermeasures. Barni and Tondi (2013) extended this analysis to cases where the adversary knows the source statistics only through training data, highlighting the complex interplay between forensic and counter-forensic tactics.

CHAPTER – IV

JOURNEY OF OBJECT DETECTION



Object Detection is vital field in computer vision that helps detect objects like cars, weapons, and people within Images or videos. This when applied in the CCTV footages it can help detect weapons, suspects, cars and many more real-time to prevent any dangerous incident. This technique recognises, localizes and also detects multiple objects simultaneously. The application of this technology spans across various domains such as health care, security, surveillance, and advanced driver assistance systems. Humans possess a native ability to quickly and precisely detect objects, identifying multiple items in complex scenes with minimal mindful effort. In contrast, computer systems require considerable data, advanced algorithms, and powerful hardware to achieve similar performance. Modern advancements in technology, such as large datasets and high-performance GPUs, have significantly improved computer vision capabilities.

Why YOLO algorithm is used in our research?

YOLO (You Only Look Once) is superior to other object detection algorithms due to its real-time performance, single-stage detection approach, and high accuracy. Unlike traditional methods that use two-stage detectors (e.g., R-CNN), which separate region proposal and classification, YOLO combines these steps into a single network, significantly reducing inference time. This makes YOLO highly efficient, capable of processing images in real-time while maintaining competitive precision and recall. Additionally, YOLO's ability to predict multiple bounding boxes and class probabilities directly from full images, rather than using a sliding window or region proposals, enhances its robustness and versatility in various applications, particularly where speed and accuracy are critical.

Digital Image Processing (DIP)

DIP involves manipulating digital images through computational techniques to enhance or extract information. A digital image is represented as a 2D function $f(x,y)$, where x & y are spatial coordinates, and the function value represents the image's intensity or color at those coordinates.

DIP is categorized into three levels of processing:

Low-Level Processing: Involves in noise reduction, contrast enrichment, and image sharpening with both Inputs & outputs being Images.

Mid-Level Processing: Includes segmentation, edge detection, and object extraction where the Inputs are images and the outputs are properties derived from Images.

High-Level Processing: This task understands and interprets images. Including complete scene and image analysis. This level involves insightful tasks associated to human vision.

Why Image Processing?

Image processing is essential for preparing digital images for viewing and enhancing their appearance. It includes:

- **Image-to-Image Transformations:** Modifying images directly.
- **Image-to-Information Transformations:** Extracting meaningful information from images.
- **Information-to-Image Transformations:** Converting information back into images.

Pixel and Resolution

- **Pixel:** The smallest unit of an image, representing the light intensity at a specific location. In an 8-bit grayscale image, pixel values range from 0 to 255.
- **Resolution:** It includes pixel resolution (total number of pixels), spatial resolution, temporal resolution, and spectral resolution. Higher resolution generally means better image quality but can be costly to achieve. Techniques like Super Resolution can enhance low-resolution images.

Gray Scale and Color Images

- **Gray Scale Image:** Represented by a single intensity function $I(x,y)$, where x and y are spatial coordinates.
- **Color Image:** Represented by three functions for the primary colors (R, G, B). Conversion to digital form involves sampling coordinates and quantizing intensity values.

Techniques in Object Detection

Object detection involves many techniques such as feature based object which rely on manually crafted features like edges, corners, and textures. The Viola-Jones

algorithm uses Haar-like features and a cascade of classifiers for real-time object detection, mainly face detection. The SVM Support Vector Machines (SVM) combined with Histogram oriented gradients (HOG) features provide a robust object detection. The Deep learning object detection models like YOLO, Faster R-CNN, and SSD (Single Shot MultiBox Detector) has revolutionized the entire field of object detection. These methods learn features automatically from massive datasets and provide results with high accuracy and efficacy by using a complex Convolutional Neural Networks (CNNs).

Deep Learning

This area of machine learning trains an object detection model using techniques that mimic how the human brain functions. In order to effectively identify, localize, and classify, artificial neural networks (ANNs) need to be trained to study and learn from larger data sets. In these networks, the phrase "Deep" refers to multiple layers of neurons that process information similarly to the human brain. These deep learning models work by sending data across several networked node levels. In order to boost the model's ability to recognize complex patterns, each layer enhances the input data.

Neurons and Layers

Neural networks can be artificial, made to solve AI problems, or biological, made of actual neurons. Weights are used to model the connections between neurons, and these connections can be either positive (excitatory) or negative (inhibitory). An activation function regulates the output, as the network processes the inputs. Deep Neural Networks have several layers between input & output layers, allowing them to model complex relationships.

A neuron or node in deep learning takes in input data, transforms it, and then sends the output to the layer above. This technique is comparable to how neurons in the human brain work. Layers are used to organize neurons. The first layer is responsible for receiving data and converting it into a numerical format that can be processed. The second type of layer is called hidden layers, which serve as intermediary layers and process the input through different calculations. The output layer, which comes in third, generates the finished product using the model's training data.

Each neuron in a layer generates an output by applying a mathematical function to the weighted sum of its inputs. Following layer receives this output after that. Objective here is the reduction of the error between the model's predicted & the actual values by weights adjustment during the training.

Activation Functions

When it comes to deciding whether or not a neuron should fire, activation functions are essential. They give the model non-linearity, which helps it pick up intricate patterns. Typical activation functions consist of:

- **Threshold Function:** A binary function that becomes active when an input value surpasses a predetermined level.
- **Sigmoid Function:** Frequently employed in binary classification applications, this function maps input values to a range between 0 and 1.
- **Hyperbolic Tangent Function:** Offers superior performance in some situations when compared to the sigmoid function, mapping input values to a range between -1 and 1.
- **Rectified Linear Unit:** This frequently used unit outputs zero for negative inputs and the input value itself for positive inputs. It is efficient and simple.

Learning and Training

Deep learning models learn by changing the weights of connections between neurons to minimize the error in predictions. This process involves three basic stages, first is **Forward Propagation where the** Input data is passed through the network, layer by layer, to produce an output. The second is **Calculating the loss where** the model's output is compared to the actual target value using a loss function. The last is the **Back propagation, here the** error is propagated back through the network, & weights are adjusted to reduce the loss. The model cycles through these steps, continuously refining its weights and improving its performance.

Types of Neural Networks

- **Feed forward Neural Networks:** Data flows in one way from input to output, used for jobs like image classification.

- **Recurrent Neural Networks (RNNs):** Include feedback loops, allowing them to handle sequential data.
- **CNNs:** Specialized for processing grid-like data such as pictures, using convolutional layers to recognize features.

CNNs Architecture

CNNs generally consist of three types of layers:

1. **Convolutional Layers:** It use filters to convolve the input image and extract features. Each neuron inside a feature map is linked to a specific area of the input. The depth, stride, and zero-padding are key hyperparameters that affect the output volume.
2. **Pooling Layers:** It decrease the feature map's spatial resolution and spatial invariance is achieved. Max-pooling is commonly used for retaining the maximum value from every receptive field, reducing dimensionality and computational complexity.
3. **Fully Connected Layers:** These layers interpret features extracted by previous layers & perform high-level reasoning. The final output is usually generated by applying a softmax function or other classifiers.

CNN Training

Training CNNs involves adjusting parameters to minimize error using algorithms like back propagation. Over fitting is a common task, where the model achieves good performance on training data but bad on unknown data. Regularization techniques help moderate this issue.

R-CNN

R-CNN combines region proposals with CNNs to detect objects. It generates potential bounding boxes and classifies each using a CNN.

Faster R-CNN

The newest object identification model, Faster R-CNN, outperforms its predecessors with the integration of a Region Proposal Network into the architecture. A Fast R-CNN detector uses the high-quality region proposals. to categorize and improve item bounding boxes. Faster R-CNN greatly improves speed and accuracy by merging the

RPN with the detection network, enabling real-time object detection in images. It is a well-liked choice for a variety of computer vision jobs because of its effectiveness and performance.

SSD

SSD discretizes the output space of bounding boxes into default boxes of various aspect ratios and scales. It predicts object presence and refines box adjustments in a single forward pass through the network.

AlexNet

AlexNet is a pioneering CNN architecture with 5 convolutional layers, 3 fully connected layers, and 1 softmax layer. It was influential in demonstrating the effectiveness of deep learning for image classification.

AlexNet is a pioneering CNN architecture with 5 convolutional layers, 3 fully connected layers, and 1 softmax layer. It was influential in demonstrating the effectiveness of deep learning for image classification.

VGG

VGG is another CNN architecture known for its simplicity and effectiveness in image classification, utilizing deep convolutional networks with small receptive fields.

MobileNets

MobileNets are designed for mobile and edge devices, using depth-wise separable convolutions to reduce computational cost while maintaining accuracy.

TensorFlow

TensorFlow is an open-source library for numerical computation and machine learning, developed by Google. It supports constructing, training, and deploying object detection models with pre-trained models available for datasets like COCO and KITTI.

Caffe Framework

Overview

Caffe is a deep learning framework designed for speed, modularity, and openness. It supports various models and optimizations, making it suitable for tasks like object

detection, semantic segmentation, and more. Caffe allows easy switching between CPU and GPU, which accelerates training times.

Caffe Models

- **OpenPose:** Detects human body, hand, and facial keypoints.
- **Fully Convolutional Networks (FCNs):** Used for semantic segmentation.
- **CNN-vis:** Generates images using CNNs.
- **Speech Recognition:** Implements speech recognition with Caffe.
- **DeconvNet:** For semantic segmentation.
- **Coupled GAN (CoGAN):** For generating paired images.
- **SegNet:** Real-time semantic segmentation architecture.
- **Deep Hand and DeepYeast:** Pre-trained models for specific tasks.

Python vs. Other Languages for Object Detection

Python is preferred for object detection due to its readable code, support for libraries like NumPy, and ease of use. It offers multiple libraries for machine learning and computer vision, making it a practical choice for implementing and testing algorithms.

OpenCV

A powerful, open-source library for computer vision and machine learning applications is called OpenCV (Open Source Computer Vision Library). It aims to speed machine perception in commercial products and provide a standardized infrastructure for computer vision applications. Businesses can simply utilize and change the code thanks to its BSD license. It was formally introduced by Intel Research in 1999 with the goal of advancing CPU-intensive applications. OpenCV offers an extensive collection of features for real-time computer vision applications, supporting multiple platforms and offering interfaces for MATLAB, C++, Python, and Java. With support for MMX and SSE instructions, the library is specifically designed for real-time vision applications and runs on Windows, Linux, Android, and macOS.

Haar Cascade Classifier

It is used for object detection in images. It works as follows:

- **Training:** Requires positive (face images) and negative (non-face images) samples. Features are extracted using Haar-like features, that compute the difference in pixel intensities.
- **Feature Calculation:** Uses integral images to speed up the computation. Each feature involves a simple sum of pixel intensities within rectangular areas.
- **Classifier Cascade:** Features are organized in stages, discarding non-face windows early in the process to improve efficiency. The final classifier combines weak classifiers into a strong one to improve detection accuracy.

YOLO

SSD, R-CNN and Fast R-CNN are some of the several object detection techniques. In 2015 Joseph Redmon, along with his colleagues introduced YOLO which detects different objects in an image. In 2016 it was officially discovered and it reframed the object detection with single regression. This unified model is able to predict numerous bounding boxes and class probabilities simultaneously.

Since its release, YOLO algorithm has outperformed many ongoing algorithms in terms of speed and accuracy to detect objects. The YOLO method can be utilised for a variety of Computer Vision (CV) activities, such as those involving animals, Ariel images, the military, autonomous vehicles, sports, hospitals, and others etc.

Numerous additional variations of YOLO have been created over time, including YOLOv1, YOLOv2, YOLOv3, YOLOv4, YOLOv5, YOLOv6, YOLOv7, and YOLOv8. Prior studies have shown that speed and accuracy of YOLOv5 is better than earlier versions. The latest version of YOLO, YOLOv8, is very efficient.

The base YOLO model processes pictures in real-time at 45 FPS. It's smaller version: Fast YOLO, processes an astounding 155 FPS while still achieving double the mAP of other real-time detectors. Compared to state-of-the-art detection systems, YOLO makes more localization errors but is less likely to predict false positives on background (Redmon et al., 2016).

A novel method for object detection called YOLO was introduced by Redmon et al. (2016). Complete training and real-time speeds are made possible by the YOLO,

which maintains a high average precision. The input image is divided into a $S \times S$ grid by the system. An object's center falls into a grid cell, and that grid cell is in charge of detecting it. B bounding boxes and confidence scores for those boxes are predicted for each grid cell. The model's level of confidence that the box contains an object and its estimate of the box's accuracy in terms of predictions are both reflected in these confidence ratings. The difference between the expected and ground truth objects is the measure of confidence. In the event that there is nothing in that cell, the confidence ratings ought to be 0.

If not, the intersection over union (IOU) between the predicted box and the ground truth should be equal to the confidence score. Five predictions make up each bounding box: x , y , w , h , and confidence. The centroid of the box with relation to the grid cell's boundaries is represented by the (x, y) coordinates. The projected width and height in relation to the entire image are represented by the letters w and h . Ultimately, the IOU between the projected box and the ground truth box is represented by the confidence prediction.

The GoogLeNet model for image categorization served as the model for the network design. There are two completely linked layers in the network after 24 convolutional layers. 1×1 reduction layers are employed with 3×3 convolutional layers in place of the inception modules used by GoogLeNet. In Figure 4.1, the YOLO architecture is displayed.

Background of YOLO

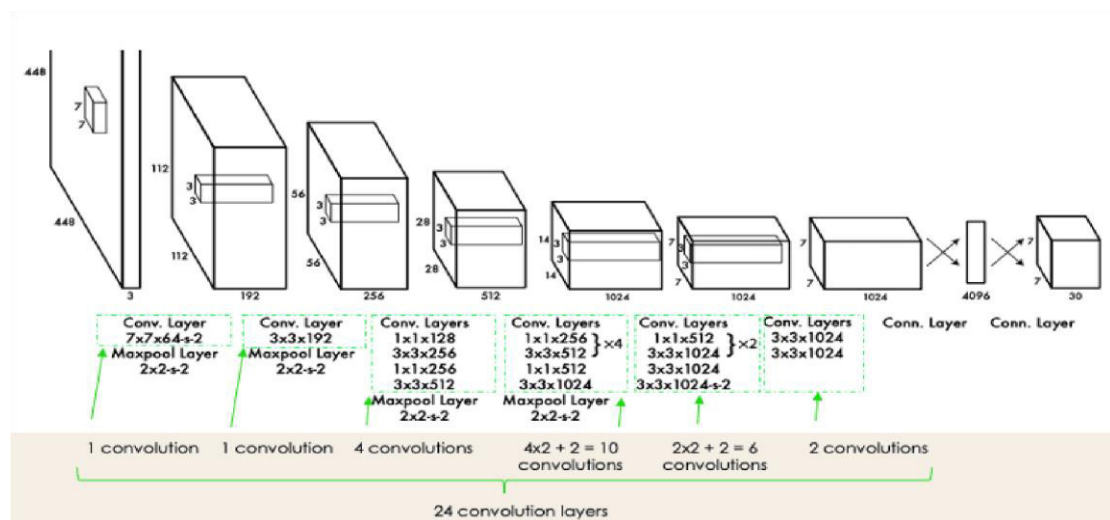


Fig. 4.1: YOLO Architecture (Redmon et al., 2016)

As further research was conducted to improve detections, the YOLO architecture was further improved by the inclusion of techniques and procedures to increase accuracy, decrease network size, and provide faster detections. These enhancements are enumerated in Table 4.1.

Improvements on YOLO Versions

From the introduction of the YOLO, there have been various changes and improvements which resulted in several versions of YOLO from YOLOv1 to YOLOv7. The findings on the YOLO versions are summarized in Table 4.1.

Table 4.1: Summary of Improvements on YOLO¹

Sr. No.	YOLO Variant	Improvement	Results
1	YOLOv1 (Redmon et al., 2016)	SSD solves the problem of drawing boundary boxes and class identification.	Higher accuracy and speed compared to Faster R-CNN
2	YOLOv2 (Redmon & Farhadi, 2018)	Iterative improvements on Batch Normalization, higher resolution detection and use of anchor boxes	Architecture reduction, quicker and more accurate identification, and improved high-resolution image detection
3	YOLOv3 (Redmon & Farhadi, 2018)	Bounding box predictions now include an objectless score, additional backbone network layer connections, and predictions at three different granularities.	Improves detection of smaller objects
4	YOLOv4 (Alexey et al., 2020)	Enhanced feature aggregations, utilization of mish activation, and a bag of	Achieved improved accuracy and ease of training, high quality

¹ Olorunshola, O. E., Irhebhude, M. E., & Evwiekpaefe, A. E. (2023). A comparative study of YOLOv5 and YOLOv7 object detection algorithms. *Journal of Computing and Social Informatics*,

Sr. No.	YOLO Variant	Improvement	Results
		freebies with mosaic augmentations	performance and accessibility
5	YOLOv5 (Nepal & Eslamiat, 2022)	Reduced network parameters, use of Cross Stage Partial Network (CSPNet) for the head, PANet for the neck of the architecture, residual structure and auto-anchor. It also utilizes mosaic augmentations.	Extremely easy to train, inference on individual, batch images, video feed and webcam ports. Ease of transfer and use of weights. Faster and more lightweight than previous YOLO.
6	YOLOv6 (Chuyi et al., 2022)	Redesigned network backbone and neck to EfficientRep Backbone and Rep-PAN Neck. The Network head is decoupled separating different features from the final head	Improvement in detecting small objects, anchor free training of model. Less stable and flexible as compared to YOLOv5.
7	YOLOv7 (Wang et al., 2022)	E-ELAN-based layer aggregation, trainable goodie bag, and 35% fewer network parameters. Model scaling for models based on concatenation.	Increase in speed and accuracy, ease of training and inference.
8	YOLOv8 (Bochkovskiy et al., 2023)	Enhanced model architecture with a focus on reducing computational cost while maintaining accuracy. Integration of advanced data augmentation techniques.	Significant reduction in computational resources needed, with comparable or improved accuracy over previous versions. Improved real-time performance for embedded systems.

The primary distinctions between the architectures of YOLOv1, YOLOv2, YOLOv3, YOLOv4, and YOLOv5, according to Nepal and Eslamiat (2022), are that YOLOv1 employs the softmax function, whereas YOLOv2 has a better resolution classifier, higher accuracy, and higher efficiency than YOLOv1. It's because the CNN of YOLOv2 has a batch normalizing layer added to it. To fetch features from input image with more efficiency & detection performance, YOLOv3 employs Darknet53 as its fundamental backbone. Multi-object classification is available in YOLOv3, meaning that an object may simultaneously fall under more than one category.

To ascertain the likelihood that an input image corresponds to a certain label, YOLOv3 substitutes an independent logistics function for the softmax function. Additionally, YOLOv3 employs the 2-class entropy loss for every category, which reduces the computational complexity caused by softmax functions.

The backbone of the YOLOv4 design is CSPDarknet53, a hybrid of the CSP and Darknet53 networks. YOLOv4 has less hardware requirements and is more accurate and efficient at detecting objects. CSPDarknet53 serves as the backbone of the Focus structure used by YOLOv5. In YOLOv5, the Focus layer first appeared. The YOLOv3 algorithm's first three layers are swapped with the Focus layer. Using a Focus layer has the advantages of requiring less memory for CUDA, having a smaller layer, and having more forward & backward propagation.

YOLOv5

A researcher Glenn and his team produced YOLOv5, a new version of the YOLO family, a month after YOLOv4 was made available. Compared to YOLOv4, YOLOv5 is about 90% lighter and faster. Using RepVGG Style structure, EfficientRep Backbone, Rep-PAN Anchor-free paradigm, SimOTA algorithm, and SIOU bounding box regression loss function, YOLOv6 has many improvements in its backbone, neck, head, and training strategies. On the other hand, YOLOv7 outperforms all other known object detectors in terms of speed and accuracy in the range of 5 FPS to 160 FPS, and has the highest accuracy (56.8% AP) of all known real-time object detectors with 30 FPS or higher on GPU V100.

With a 50% reduction in computation and 40% reduction in parameters compared to the state-of-the-art real-time object detector, YOLOv7 significantly increased real-time object detection accuracy without raising the cost of inference. It also boasts a faster inference speed and higher detection accuracy.

Nepal & Eslamiat (2022) state that YOLOv5 differs from earlier releases in that PyTorch is used in place of Darknet. It makes use of CSPDarknet53 as its mainframe. To improve the information flow, it employs the Path Aggregation Network (PANet) as the neck. PANet uses a novel feature pyramid network (FPN) with many top-down and bottom-up layers. This enhances the model's low-level feature propagation. PANet increases the localization accuracy of the item by improving the localization in lower layers.

Furthermore, YOLOv5 shares the same head as YOLOv4 and YOLOv3, which provide three distinct feature map outputs in order to accomplish multi-scale prediction. The following is a summary of the YOLOv5 model: Support structure: CSP network, focus structure, Neck: PANet, SPP block, Head: GIoU-loss YOLOv3 head. The use of CSPDarknet53, which addressed the issue of repeating gradient information present in YOLOv4 and YOLOv3, allowed YOLOv5 to outperform YOLOv4 in terms of accuracy while decreasing the network's parameters and speed of inference. Figure 4.2 displays the YOLOv5 architecture.

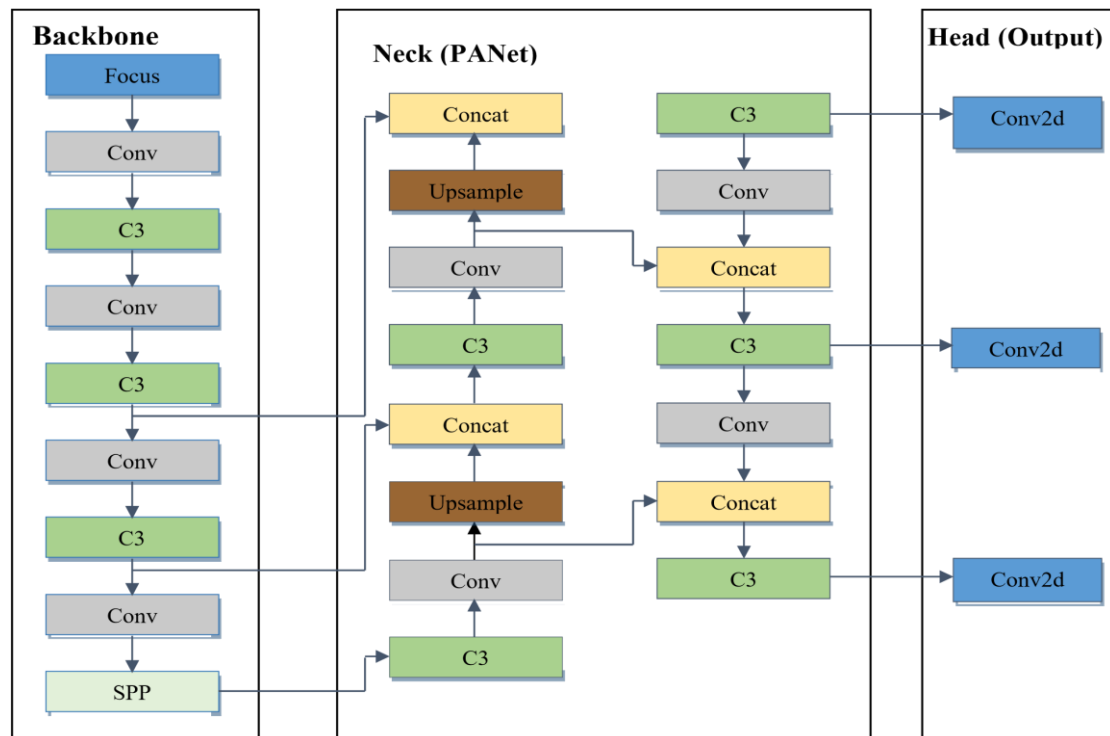


Fig. 4.2: YOLOv5 Architecture (Nepal & Eslamiat, 2022)

YOLOv7

YOLOv7 with its amazing features, YOLOv7 is a real-time object detector that is now changing the CV business. Without the use of any other datasets or pre-trained

weights, YOLOv7 was only trained from scratch on the MS COCO dataset (Wang et al., 2022). According to Wang et al. (2022), YOLOv7 has the best accuracy at 56.8% AP among all known real-time object detectors with 30 FPS or above on Graphics Processing Units (GPU) V100. It also outperforms all other known object detectors in the range of 5 FPS to 160 FPS.

With around 40% fewer parameters and 50% less processing than the state-of-the-art real-time object detector, YOLOv7 significantly increased real-time object detection accuracy without increasing inference costs. It also boasts faster inference speed and higher detection accuracy.

Extended efficient layer aggregation networks (E-ELAN) are a feature of YOLOv7. To continuously improve the network's capacity for learning without erasing the initial gradient path, E-ELAN employs the expand, shuffle, and merge cardinality techniques (Wang et al. 2022). The architecture of the transition layer remains unaltered, while E-ELAN solely modifies the architecture of the computing block. Apart from preserving the original E-LAN design architecture, E-ELAN also helps various computational block groups to acquire a wider range of characteristics.

Model scaling for concatenation-based models is also available in YOLOv7. Model scaling is mostly used to modify certain model features and produce models at various scales in order to accommodate varying inference speeds. The suggested compound scaling approach can preserve both the ideal structure and the characteristics that the model possessed at the time of its original design. The model scaling for YOLOv7 concatenation-based models is shown in Figure 4.3.

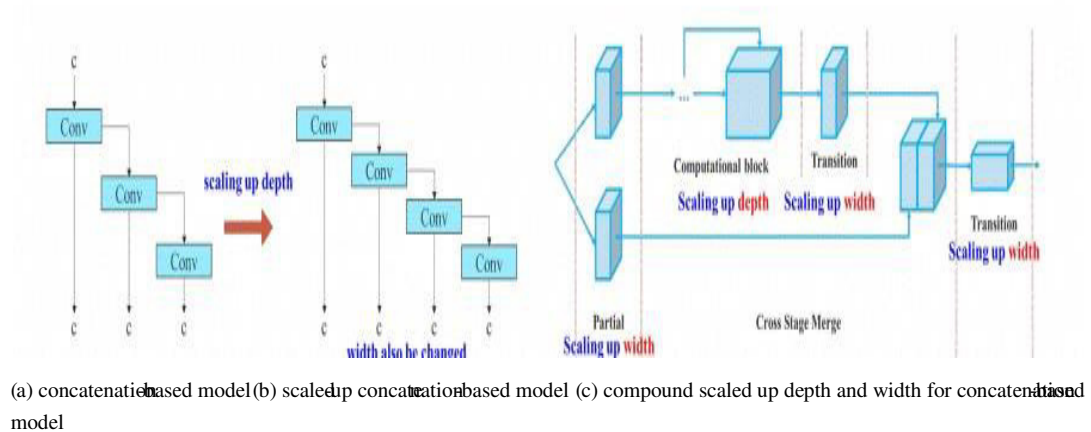


Fig. 4.3: Model Scaling of YOLOv7 (Wang et al., 2022)

From (a) to (b), it can be shown that the output width of a computational block likewise grows when depth scaling is applied to concatenation-based models. The next transmission layer's input width will rise as a result of this phenomena. Thus, (c) is suggested: for concatenation-based models, model scaling entails scaling only the depth of a computing block and scaling the appropriate breadth of the remaining transmission layer.

When compared to YOLOv5 and YOLOv7, YOLOv6 offers even greater advances in terms of detection, but it is less scalable and requires more effort to train. Additionally, YOLOv6 outperforms YOLOv5 and YOLOv7 in terms of accuracy when utilized for single image inference as opposed to multiple image inference (Banerjee, 2022). Because they are suitable for multiple item detection and make it simple to customize the training and inference process, YOLOv5 and YOLOv7 were used in the experiment.

YOLOv8

YOLOv8 is the latest advancement in the YOLO (You Only Look Once) series of real-time object detectors, building on the strengths of its predecessors to achieve higher accuracy and faster inference speeds. It features an optimized backbone and neck architecture that enhances the detection of objects of varying sizes and shapes, particularly improving the accuracy for small and densely packed objects. YOLOv8 also introduces improved model scaling, allowing it to adjust depth, width, and resolution to balance speed and accuracy depending on the hardware available, while maintaining the model's structural integrity and feature representation. One of the key innovations in YOLOv8 is dynamic label assignment, which dynamically adjusts ground-truth labels during training based on the network's predictions, improving the model's focus on hard-to-detect objects. Additionally, advanced training techniques, including enhanced data augmentation and improved loss functions, contribute to better generalization and robustness against overfitting. YOLOv8 continues to optimize inference speed, making it well-suited for real-time applications, even on resource-constrained devices. Incorporating features like Cross-Stage Partial Networks (CSPNet), YOLOv8 reduces model size and computational complexity while retaining high representational capacity. Designed to surpass the performance benchmarks of YOLOv7 and other contemporary detectors, YOLOv8 represents a significant leap forward in the field of real-time object detection, maintaining a strong balance between speed and accuracy.

Applications of Object Detection

Facial Recognition

Facial recognition systems, such as "Deep Face" by Facebook and Google's facial recognition in Photos, identify and verify human faces using deep learning techniques. They analyze facial features like eyes, nose, and mouth for accurate recognition.

People Counting

Object detection aids in counting individuals for purposes such as security, store performance analysis, and crowd management. Challenges include handling fast-moving individuals and varying crowd densities.

Industrial Quality Check

In industrial settings, object detection improves processes like sorting, inventory management, and quality control by automating the identification and classification of products.

Self-Driving Cars

Self-driving cars use object detection to perceive their surroundings, integrating radar, LIDAR, GPS, and computer vision to navigate and avoid obstacles autonomously.

Security

Object detection enhances security through applications like facial recognition, retina scans, and real-time monitoring, aiding in criminal identification and surveillance.

Object Detection in Video Surveillance

In video surveillance, object detection is crucial for monitoring and analyzing video feeds. The process typically involves:

- **Pre-Processing:** Enhancing video quality and reducing noise.
- **Segmentation:** Dividing frames into meaningful regions to isolate objects.
- **Foreground and Background Extraction:** Identifying moving objects against a static background.
- **Feature Extraction:** Analyzing object characteristics for classification and tracking.

Effective video surveillance relies on robust object detection to handle challenges like varying lighting conditions and occlusions.

CHAPTER – V

EXPERIMENTAL SETUP AND EVALUATION



Experiment Setup

In order to conduct the experiment a powerful GPU and TPU was needed to train, validate and a custom machine learning model. In order to do test this custom based model google Colab a free resource was used. The data set was uploaded to google drive that was then accessed by Google Colab environment. The Google Colab provided Python programming capabilities with access to popular deep learning libraries.

The model was trained using individual validation dataset to evaluate its performance, efficacy and generalization ability. To assess the performance various metrics such as accuracy, precision, recall, F1-score, mean average precision (Map) were employed. Google drive was used as a primary storage facility to store pre-processing and training data to reduce manual intervention for storage. The experimental data such as weights, evaluation metrics, and visualizations were redirected to be stored on the google drive for easy and secured analysis for future use.

5.1 Advantages of Google Colab for Experimentation

Google Colab offers free computing resources with certain limitations, making it affordable for researchers and developers. The users can leverage GPUs and TPUs for quicker calculation, enabling faster model training and experimentation. Another advantage of the colab notebooks is that the notebook can easily be shared and collaborates. This fosters teamwork and knowledge sharing.

5.2 Dataset Preparation

A dataset was prepared and classified into two groups, of five classes consisting of objects and five classes of people.

Object Classes

Images of the object classes the images were sourced from Google Open Images Dataset and converted into the YOLO (You Only Look Once) format using the OIDv4 Toolkit. This Python based package facilitated the extraction of specific parts of the Open Image dataset to create custom object dataset to create object such as Knife, Handgun, Bottle, Axe, and Hammer.

First, a repository clone was used to download the OIDv4 Toolkit.

This required a few steps: installing Anaconda and setting up and activating a virtual environment came first.

After that, Git was installed to make repository cloning possible.

The repository was cloned into the specified directory using the following command:

```
git clone https://github.com/EscVM/OIDv4_ToolKit.git
```

Next, the directory was changed to the cloned OIDv4_ToolKit folder, and the necessary dependencies were installed using:

```
pip install -r requirements.txt
```

Pandas, numpy, awscli, urllib3, tqdm, and opencv-python were among these dependencies. After the setup was finished, we downloaded the pictures for the classes we had designated. The following command was run in order to accomplish this:

```
python main.py downloader --classes Knife Handgun Bottle Axe Hammer --  
type_csv train --multiclass 1 --limit 200
```

The classes to be downloaded are specified by the `classes` option in this command, and the `--type_csv train` option indicates that training data is being downloaded.

To guarantee that every image is downloaded into a separate folder, use the `--multiclass 1` option.

A limit of 200 photos per class is imposed by the `--limit 200` argument.

Thus, the goal was to download a maximum of 1000 photos using five classes.

The actual number of photographs acquired, however, may have been slightly lower because some may have been deleted from the original website.

This structured approach enabled us to efficiently prepare a robust dataset for our experimental setup, ensuring the integrity and organization necessary for effective object detection model training. The number of object images downloaded are shown in Table no 5.1.

Table 5.1 : Class and no of images downloaded

CLASS	NO. OF IMAGES
Knife	200
Handgun	200
Bottle	200
Axe	115
Hammer	114

Pre-existing labels and bounding boxes that identified the objects in each image were included with these pre-annotated photographs. This removed the requirement for manual labelling, which may be a labour- and time-intensive procedure. In manual labelling, each image is reviewed by a human annotator who then manually draws bounding boxes around objects of interest and labels them with the appropriate class. It takes a lot of work to complete this process.

We expedited the process of preparing our dataset by utilizing pre-annotated photos, which guaranteed high-quality and consistent annotations. Furthermore, because the annotations were consistent and applied to every image, the usage of pre-annotated data made the training process for our object detection model more effective and dependable. Maintaining consistency is essential for training reliable and accurate models because it guarantees that the model gains knowledge from appropriately labelled input, which enhances its performance and capacity for generalization.



Fig. 5.1 : csv_folder and Dataset folder created



Name	Type	Size
 class-descriptions-boxable	Microsoft Excel Co...	11 KB
 train-annotations-bbox	Microsoft Excel Co...	11,66,049 ...

Fig. 5.2 : class-descriptions-boxable.csv and train-annotations-bbox.csv files created in csv_folder

After executing the above steps, 2 files were created in csv_folder i.e. class-descriptions-boxable.csv and train-annotations-bbox.csv as displayed in figure 5.1 and 5.2.

The class-descriptions-boxable.csv displayed in figure 5.3 contains the name of all the classes with their corresponding 'LabelName' and the validation-annotations-bbox.csv file contains one bounding box (bbox for short) coordinates for one image,

and it also has this bbox's LabelName and current image's ID from the validation set of OIdv4.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	/m/011k07	Tortoise											
2	/m/011q46kg	Container											
3	/m/012074	Magpie											
4	/m/0120dh	Sea turtle											
5	/m/01226z	Football											
6	/m/012n7d	Ambulance											
7	/m/012w5l	Ladder											
8	/m/012xff	Toothbrush											
9	/m/012ysf	Syringe											
10	/m/0130jx	Sink											
11	/m/0138tl	Toy											
12	/m/013y1f	Organ											
13	/m/01432t	Cassette deck											
14	/m/014j1m	Apple											
15	/m/014sv8	Human eye											
16	/m/014trl	Cosmetics											
17	/m/014y4n	Paddle											
18	/m/0152hh	Snowman											
19	/m/01599	Beer											
20	/m/01_5g	Chopsticks											
21	/m/015h_t	Human beard											
22	/m/015p6	Bird											
23	/m/015qbp	Parking meter											
24	/m/015qff	Traffic light											
25	/m/015wgc	Croissant											
26	/m/015x4r	Cucumber											

Fig. 5.3 : class-descriptions-boxable.csv

5.3 People Classes

For the people classes, images were downloaded from Google and processed further. The classes included Bill Gates, Elon Musk, Marilyn Monroe, Leonardo DiCaprio, and Will Smith. For each classes the steps such as auto-orienting the pixel data striped the EXIF orientation metadata. The images were resized to 640x640 pixels with stretching to fit the dimensions. The Images were augmented by applying three 90-degree rotations, along with a random rotation of -15 and +15 degrees horizontally and vertically. To expose the model to varied scenario. For bill gates 61 images were pre-processed and augmented to 130 images; For the Elon musk class 151 images were augmented using 63 original images; with an initial set of 63 images of Marilyn Monroe the dataset was expanded to 144 images after augmentation; After pre-processing and augmentation of 62 images of Leonardo DiCaprio the dataset stretched to 148 images; For the final class of people images a set of 62 images of will smith were pre-processed and augmented to expand the dataset to 148 images.

We augmented the dataset of images featuring people. Implementing image augmentation techniques expanded the dataset and helped reduce overfitting. Cleaning and augmenting image data can significantly improve the model's performance.

Each of these steps ensured that the dataset was robust and suitable for training object detection models.

Labelling of Images

The images were labelled by a user-friendly platform- **Roboflow**. This powerful platform helped in efficiently labelling the images in order to get precise and consistent annotations across all images. The advanced features such as automatic annotation suggestions greatly assisted in fastening the labelling process with great efficacy. Each object with in the image was carefully annotated to match the resulting labels making it fit for training high-performance object detection models.

Also, to ensure transparency and facilitate easy access to the dataset the links to all images were maintained in an excel file, serving as a comprehensive reference, containing the URLs of each image.

This organized approach not only aids in tracking the origin of each image but also makes it easier for other researchers to verify and utilize the dataset. By providing this level of detail and accessibility, we ensure that our dataset preparation process is both transparent and reproducible, supporting the integrity and utility of our research.

Table 5.2 : Before & After Pre-processing People Image Dataset

CLASS	No. of instances Before Pre-processing	No. of instances After Pre-processing and Augmentation
BillGates	61	130
ElonMusk	63	151
MarilynMonroe	63	144
LeonardoDicaprio	62	148
WillSmith	62	148

This table 5.2 illustrates the increase in the number of instances for each class of people after applying the pre-processing and augmentation steps to the initial set of

images. The augmentation process significantly enhanced the dataset, making it more suitable for training robust object detection models.

Table 5.3 : Class instances and their no. of annotation

CLASS	CLASS INSTANCES	NO. OF ANNOTATION
Knife	200	258
Handgun	200	254
Bottle	200	509
Axe	115	148
Hammer	114	139
BillGates	130	130
ElonMusk	151	149
MarilynMonroe	144	144
LeonardoDicaprio	148	148
WillSmith	148	148

This table 5.3 details the number of instances and corresponding annotations for each class in the dataset. The variations in the number of annotations reflect the complexity and distinctiveness of each class, contributing to a comprehensive dataset for object detection model training.

Table 5.4 : Dataset Summary

	With Pre-processing and Augmentation
Total Images	1550
Classes	10
Unannotated	0
Training Set	1085(70%)
Validation Set	310(20%)
Testing Set	155(10%)
Annotation	2032(.3 per image (average))

This table 5.4 provides a summary of the dataset used for training, validation, and testing. The dataset consists of 1550 images across 10 classes, all of which have been annotated. The distribution of images ensures a balanced split for training (70%),

validation (20%), and testing (10%) purposes. On average, each image has 0.3 annotations, leading to a total of 2032 annotations.

This dataset, was further trained on the three versions of YOLO (You Only Look Once) object detection model: YOLOv5, YOLOv7, and YOLOv8. Each model underwent due training to use the pre-processed and augmented dataset to ensure consistency and robustness in the training process.

Training:

- **YOLOv5:** firstly, YOLOv5, known for its balance between speed and accuracy was trained.
- **YOLOv7:** As an improved version YOLOv7 incorporates advanced techniques to enhance detection accuracy and speed.
- **YOLOv8:** The Dataset was trained using the latest iteration YOLOv8, to leverage on latest advancements.

5.4 Evaluation Metrics

To assess the performance of our model using the created dataset, we computed Precision, Recall, and Average Precision (AP). These metrics help in evaluating the classification accuracy, detection ability, and overall effectiveness of the model. The following formulas were used to calculate these metrics:

Precision (P)

Precision measures the accuracy of the classification by determining the ratio of correctly identified positive instances to the total number of positive instances identified. It indicates how many of the identified instances are actually relevant. The formula for Precision is:

$$Precision = \frac{True\ Positive}{(True\ Positive + True\ Negative)}$$

Recall (R)

Recall measures the ability of the model to correctly identify all relevant instances within the dataset. It is the ratio of correctly identified positive instances to the total number of actual positive instances. The formula for Recall is:

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

F1-Score

The F1-Score provides a balance between Precision and Recall by considering both metrics in its calculation. It is the harmonic mean of Precision and Recall, making it a more accurate measure than accuracy alone, especially in cases of imbalanced datasets. The formula for the F1-Score is:

$$F1 - score = \left(\frac{Recall^{-1} + Precision^{-1}}{2} \right)^{-1}$$

where:

- True Positives (TP) are positive samples classified correctly.
- False Positives (FP) are negative samples classified incorrectly.
- False Negatives (FN) are positive samples classified incorrectly.

Average Precision (AP) and Mean Average Precision (MAP)

Average Precision (AP) provides the precision of the model across different recall values. It is a weighted mean of precisions achieved at each threshold, with the increase in recall from the previous threshold used as the weight. Mean Average Precision (MAP) is the average of the AP values for each class, providing a single performance measure for object detection models.

Frames Per Second (FPS)

Frames Per Second (FPS) is a crucial speed performance metric that indicates the number of images the model can process per second. Higher FPS values are desirable as they reflect the model's efficiency and speed in real-time applications.

To evaluate the performance of our model, we computed several key metrics: Precision (P), Recall (R), F1-Score, Average Precision (AP), Mean Average Precision (MAP), and Frames Per Second (FPS). Precision measures the accuracy of positive predictions, while Recall assesses the model's ability to identify all positive instances. The F1-Score provides a balanced measure between Precision and Recall. AP evaluates the precision across different recall thresholds, and MAP summarizes this performance across all classes. FPS indicates the model's processing speed in images per second. Together, these metrics offer a comprehensive assessment of the model's accuracy, detection capability, and efficiency.

CHAPTER – VI

RESULTS AND DISCUSSION



Our dataset was trained on YOLOv5 object detection model and the results of summary of performance metrics is shown below in the Table 6.1. As mentioned in previous chapter, our final dataset contains 1550 images.

6.1 YOLOv5 RESULTS

Table 6.1 : Result of YOLOv5 object detection model

CLASS	IMAGES	LABELS	P	R	MAP@0.5	MAP@0.5:0.95
all	310	480	0.37	0.246	0.0803	0.0407
Axe	310	29	1	0	0.0261	0.012
BillGates	310	32	0.107	0.469	0.113	0.0531
Bottle	310	179	0.0359	0.00559	0.0124	0.00273
ElonMusk	310	20	0.0708	0.7	0.192	0.118
Hammer	310	26	1	0	0.00839	0.00252
Handgun	310	40	0.132	0.05	0.0252	0.00768
Knife	310	54	1	0	0.0191	0.0043
LeonardoCaprio	310	29	0.116	0.724	0.167	0.0938
MarilynMonroe	310	29	0.0296	0.0345	0.0424	0.016
WillSmith	310	42	0.21	0.476	0.198	0.0968

Class Interpretations:

1. Axe:

The results of the "Axe" class shows perfect precision ($P = 1.0$) is achieved alongside zero recall ($R = 0.0$), which highlights the critical insights of model's behaviour and its inferences for practical applications.

A precision of 1.0 indicates that when the model predicts the presence of an "Axe" in an image, it is always right. This high value of precision implies a complete absence of false positive predictions for Axe class, suggesting that when the model identifies an "Axe," it is definitely present in the image. However, this remote study of precision alone does not provide a complete understanding of the model's efficacy.

Recall measures the capability of the model to detect all actual cases of a class within the dataset. Here, recall of 0.0 shows that the model fails to identify any

true instances of "Axe" present in the images. This shortfall in recall implies a serious restraint in the model's ability to broadly separate and detects "Axe" objects, which results in a high number of false negatives instances of "Axe" that are not detected by the model.

Here, precision of 1.0 with 0 recall shows a disparity in the performance of the model. It can correctly detect images that do contain axes but it supervises every true instance of an "Axe" present in the dataset which highlights a crucial trade-off between accuracy and recall that needs to be addressed for the model to be almost viable.

2. BillGates:

The precision (P) is 0.107 and a recall (R) is 0.469 for "BillGates" class. It indicates that the model predicts "BillGates" class correctly only about 10.7% of the time.

So, there is a high rate of false positive calculations where the model incorrectly detects some objects or person as "BillGates." This aspect of the model's performance presents challenges in terms of accuracy and consistency for applications relying on precise object detection.

The relatively low precision value suggests room for upgrading in the model's ability to determine true instances of "BillGates" from other visual elements within the dataset. The significance of this is potential imprecisions and deceptive results in downstream applications that utilize the model's predictions.

On the other hand, the recall value of 0.469 signifies that the model can detect nearly 46.9% of all actual occurrences of Bill Gates present in the images. This moderate recall rate indicates a capability to identify a substantial proportion of relevant instances of "BillGates," indicating a reasonable level of efficacy in capturing instances of this specific individual within the dataset.

3. Bottle:

The precision (P) of 0.0359 and a recall (R) of 0.00559 for "Bottle" class indicates that the model detects with accuracy of only 3.59%. This low precision rate indicates that the predictions made by the model are mostly false

positive. The values indicate serious shortfalls in the model's ability to detect and identify the instances of the bottle within the dataset. The precision value of 0.0359 and accuracy of 3.59% shows large number of false positive results, indicating in practical challenges in the application of the model in the real-world scenario. Ironically the recall value of 0.00559 for the object of the bottle indicates that the model can detect all instances present in the dataset with accuracy of 0.559% only.

In contract with the precision value of 0.0359 and accuracy of 3.59%, indicates high chances of false positive results. On the other hand, the recall value of 0.00559 "Bottle" class signifies that the model can detect only around 0.559 % of all accurate instances of bottles present in the dataset. This significantly low recall rate indicates the models incompetency to detect the bottles accurately sabotages the performance of the model and its usability to detect the bottles in real world scenario.

4. ElonMusk:

The precision (P) of 0.0708 and a recall (R) of 0.7 for "ElonMusk" class indicates that the model detects with accuracy of only 7.08%. This low precision rate indicates that the predictions made by the model are mostly false positive. The results indicate serious challenges in the model's ability to detect and identify the instances of the bottle with in the dataset. This precision value of 0.0708 indicates high chances of false positive results indicating the short falls of the model.

On the other hand, with the recall value of 0.7 for the class "ElonMusk", the model detects around 70% of all true instances within the dataset. This high recall value gives 70% accuracy in detecting all instances of "Elon Musk" within dataset. This shows the models stronger capability to detect this specific individual compared to other classes with lower recall rates.

This comparison of low precision with high recall for the "ElonMusk" indicates certain strength and weakness in the model's ability to detect this specified individual. Model proves a strong ability to capture true instances of Elon Musk (high recall) while suffering false positive predictions due to the

low precession rate leading to inaccuracies and certain short falls in predictions of this class.

5. Hammer:

The precision (P) of 1.0 and a recall (R) of 0.0 for “Hammer” class indicates models’ strengths and weakness in accurately identifying instances of hammers. Giving a precision value of 1.0 the model achieves a prefect precision each time it detects hammer. This perfect precision score signifies no false positive predictions for hammers, indicating in great confidence in the model’s ability to identify this specific object. On the other hand, due to the perfect precision we also need to critically examine recall metrics, to assess the model’s overall efficiency.

In contract the recall of 0 signifies that the model misses all instances of hammers present in the images, resulting in complete inability to identify this object. Despite achieving perfect precision, the lack of any recall for hammers highlights a severe flaw in in the model's detection capability for this very class.

This understanding of precision and recall metrics for the "Hammer" class reveals unevenness in the performance of the model. While the model is superior in giving perfect precision its inability to detect any true instances of hammer with a zero recall questions the model’s practicality and effectiveness in real-world applications needing an accurate object detection.

6. Handgun:

The precision (P) of 0.132 and a recall (R) of 0.05 for “Handgun” class shows that the model successfully detects Handgun 13.2% of the time. This low precision rate indicates that the predictions made by the model are mostly false positive. They do not match the actual handgun in the image. The model frequently misclassifies other objects as handguns, leading to potential inaccuracies and inadequacies in applications dependency on these predictions.

On the other hand, the recall value for “Handgun” is 0.05 resulting in only 5% detection of all true instances. This extremely low recall rate indicates that the

model misses a huge number of actual handguns present in the images. This inability of the model to capture most actual instances of handguns shows a critical limitation in its competence to detect object efficiently.

The blend of low precision and very low recall suggests that the model struggles both in precisely identifying true instances of handguns and in generously capturing and recognizing this object class within the dataset. The models reveal specific challenges and shortcomings of the model in detecting handguns precisely to the values of precision and recall metrics.

7. Knife:

The precision (P) of 1.0 and a recall (R) of 0.0 for “Knife” class indicates models’ strengths and weakness in accurately identifying instances of Knife. Giving a precision value of 1.0 the model achieves a prefect precision each time it detects Knife. This perfect precision score signifies no false positive predictions for Knife, indicating in great confidence in the model’s ability to identify this specific object. On the other hand, due to the perfect precision we also need to critically examine recall metrics, to assess the model’s overall efficiency.

In contract the recall of 0 signifies that the model misses all instances of Knife present in the images, resulting in complete inability to identify this object. Despite achieving perfect precision, the lack of any recall for Knife highlights a severe flaw in in the model's detection capability for this very class.

This understanding of precision and recall metrics for the " Knife " class reveals unevenness in the performance of the model. While the model excels in giving perfect precision), its inability to detect any true instances of Knife (zero recall) raises significant questions in its practical utility and effectiveness in real-world applications needing an accurate object detection.

8. LeonardoCaprio:

The precision (P) of 0.116 and a recall (R) of 0.724 for “LeonardoCaprio” class indicates that the model successfully detects this class object 72.4% of the time. This low precision rate indicates that the predictions made by the model are mostly false positive. They do not match the actual Leonardo

DiCaprio in the image. Many of the instances that the model identifies as Leonardo DiCaprio are incorrect, which suggests that the model is not very sharp in its detection of this particular class. This high recall suggests that the model is quite active in capturing most of the real incidences of Leonardo DiCaprio, preventing it from not missing many true instances. However, while the model is good at finding instances of Leonardo DiCaprio, it also includes a lot of improper findings. The high recall paired with low precision suggests that the model prioritizes identifying as many true instances as possible, even if it means accepting a good number of false positives.

9. Marilyn Monroe :

The precision of 0.0296 for the "MarilynMonroe" class suggests that the model's predictions for Marilyn Monroe highly inaccurate supporting it with a very high rate of false positive predictions. The results indicate that only 2.96% of the instances are correctly identified. There is a significant drawback in the model's ability to precisely differentiate Marilyn Monroe from other objects or people in the dataset

In contrast, the recall of 0.0345 indicates that the model successfully detects only a small fraction, specifically 3.45%, of all true instances of Marilyn Monroe within the dataset. Having such a low recall rate Indicates that the model misses a vast majority of the real occurrences of MarilynMonroe. This signifies a serious shortage in the models training and capability to identify and classify Marilyn Monroe accurately.

10. WillSmith :

The precision (P) of 0.21 and a recall (R) of 0.476 for "WillSmith" class indicates that the model successfully detects this class object 21% of the time. This indicates a moderate rate of false positive predictions. Consequently, the model often wrongly identifies other people or objects for Will Smith, showing its limited accuracy in differentiating.

In contract to the precision, the recall of 0.476 indicates that the model successfully detects approximately 47.6% of all true instances of Will Smith within the dataset. This shows a moderate recall indicating that the model has

brief ability to recognize and identify “WillSmith” even though it fails to detect more than half of the real instances present in the dataset. Therefore, the model displays a partial ability to detect Will Smith, but ironically its overall performance is tampered both the significant rate of false positives and its inability to dependably detect true instances, indicating a strong need in its training and accuracy.

Overall Interpretation:

The overall assessment of an object detection model indicates significant challenges across various classes, showing a gap in precision and recall metrics underlining the requirement of considerable improvements in performance. The model displays commendable precision (1.0) for specific classes such as "Axe," "Hammer," and "Knife," with complete absence of false positive results. On the contrary this precision comes at a cost of zero recall, indicating a critical failure to detect any real instance of these classes. This result is alarming as it signifies that the model is confident in its detection, however it misses the real presence of these objects in the images, highlighting a fundamental error in its ability to detect these objects.

On the other hand, for the classes like "BillGates," "ElonMusk," and "WillSmith," the model's precision remain low leading to a significant number of false positive results. This variable inconsistency in precision highlights the hurdles faced by the model, resulting in misclassifications that can impact the applications reliable on accurate object detection. Besides, recall rates across all classes are usually poor, representative that the model scuffles to detect an important portion of accurate instances for most classes, further showing shortages in its ability to largely capture the objects of interest within images.

The Mean Average Precision (MAP) scores, particularly at IoU (Intersection over Union) thresholds of 0.5 (0.0407) and across thresholds from 0.5 to 0.95 (0.012), further reinforce the overall poor performance of the model in object detection tasks. The low MAP values suggest that the model's capability to accurately localize and detect objects across various classes is inadequate, especially under more stringent IoU thresholds, which are crucial for ensuring precise localization of objects within images.

Time_100_Will_Smith_a.jpg	ElonMusk0.3.jpg	Reunion_con_Leonardo_DiCaprio_Musical_the_Neural.jpg	ElonMusk0.3.jpg	ElonMusk0.3.jpg
Nobel_Peace.jpg	ElonMusk0.3.jpg	ElonMusk0.3.jpg	ElonMusk0.3.jpg	ElonMusk0.3.jpg
Marilyn_Monroe_figure_at_Madison_Square_Garden.jpg	ElonMusk0.3.jpg	ElonMusk0.3.jpg	ElonMusk0.3.jpg	ElonMusk0.3.jpg

Fig. 6.1 : Result of YOLOv5 model on Image

6.2 YOLOv7 RESULTS

Table 6.2 : Result of YOLOv7 object detection model

CLASS	IMAGES	LABELS	P	R	MAP@0.5	MAP@0.5:0.95
all	310	480	0.435	0.474	0.432	0.238
Axe	310	29	0.307	0.0345	0.175	0.0769
BillGates	310	32	0.357	0.831	0.576	0.319
Bottle	310	179	0.463	0.145	0.144	0.0503
ElonMusk	310	20	0.287	0.95	0.712	0.425
Hammer	310	26	0.287	0.269	0.158	0.0637
Handgun	310	40	0.461	0.342	0.325	0.121
Knife	310	54	0.456	0.296	0.296	0.152
LeonardoCaprio	310	29	0.367	0.724	0.523	0.357
MarilynMonroe	310	29	0.75	0.722	0.826	0.442
WillSmith	310	42	0.62	0.429	0.588	0.371

The results of summary of YOLOv7 model's performance across different classes is displayed in Table 6.2. It reveals insights into its efficiency in detecting specific objects within the dataset. Analyzing the precision, recall, and Mean Average Precision (MAP) scores for each class provides an inclusive understanding of the model's strengths and weaknesses.

Class-Specific Interpretation:

1. Axe:

The performance of YOLOv7 model in detecting the object “Axe” was below average due to a low precision rate of 0.307, indicating successful detection of this class object 30.7%. of the time. Such a low precision indicates a significantly high rate of false positive predictions, where the model falsely identifies other objects as axes as well.

On the other hand, the recall for axes is significantly low at 0.0345, indicating that the model only captures 3.45% of all true instances of the object of axes within the dataset. The model certainly misses vast majority of real axes present in the dataset. This proves incompetency of the model to effectively

detect axes, compromising its usefulness in tasks requiring precise identification of this object class.

Overall, the YOLOv7 model's performance comprises of high rate of false positive predictions and inability to detect all true instances of the object “axe”. Suggesting a substantial limitation and need for targeted improvements in the data quality, model architecture and training strategies to enhance precision and reliability in detecting “axe” objects class.

2. Bill Gates:

The YOLOv7 model displays a balanced performance in detecting instances of Bill Gates with a moderate precision score of 0.357 and successfully detects the images of Bill Gates 35.7% of the time. This precision value suggests in some false positive predictions with reasonable accuracy in detecting all true instances of Bill Gates.

The model shows a high recall rate of 0.831 for Bill Gates. This recall value indicates that the model successfully detects 83.1% of all instances of Bill Gates with in the dataset. This high recall shows that the model is competent to detect all majority of real instances of Bill gates, with a low false negative detection.

Overall, the YOLOv7 model's performance for Bill Gates shows efficiency of the model to detect the object. The moderate precision score indicates some presence of false positives. In contract to the high recall rate indicating the accuracy and efficiency of the model to detect bill gates, making this suitable for practical application of the model in real world.

3. Bottle:

The YOLOv7 model demonstrates a mixed performance in detecting instances of bottles with a high precision score of 0.463 and successfully detects the images of this class object 46.3% of the time. This indicates a low rate of false positives for the class bottle.

The model has a low recall rate of 0.145 indicating that the model only captures 14.5% of all true instances. On the other hand, the model misses a

significant proportion of real bottle instances, leading to incomplete identification.

The high precision score indicates that the predictions of the model are accurate, with a low rate of false positives. However, the low recall rate indicates that the model misses a considerable number of instances.

However, the YOLOv7 model demonstrates both accuracy and efficacy in detecting the bottles with a high precision rate. In contrast its performance is limited due to a low recall rate. Improvements in the model's recall to detect the object would enhance the efficacy in detecting this object class, ensuring reliability of the model in real world application.

4. ElonMusk:

The YOLOv7 model's performance in detecting instances of Elon Musk gives mixed results with a precision score of 0.287 and successfully detects the images of this class object 28.7% of the time, indicating a high rate of false positive predictions. The model incorrectly detects Elon Musk.

Although the model has low precision rate it displays a high recall score of 0.95 for Elon Musk, indicating that the model successfully detects 95% of all true instances of Elon Musk. The high recall suggests the efficacy of the model in detecting all instances of Elon Musk. This high recall rate signifies model's ability to detect Elon Musk. However, the low precision indicates a significant number of false positives leading to inaccuracies and inadequacies in real world application to detect Elon Musk.

Overall the YOLOv7 model demonstrates strong efficacy to detect Elon Musk due to a high recall rate on the other hand its performance is limited by low precision rate. The model needs improvement in reducing false positive predictions. Enhancement in precision would increase the model's accuracy and reliability in identifying Elon Musk, making it more able for practical usage in real world scenarios.

5. Hammer:

The YOLOv7 model's performance shows a balance between precision and recall to detect hammer. The precision score is 0.287 which indicates this model successfully detects the images of this class object 28.7% of the time, suggesting a moderate false positive rate for this class, some instances are wrongly identified as hammers.

The low recall score for hammers which is 0.269 indicates that it detects with an efficacy of only 26.9% of hammer within the dataset. This low recall suggest that the model misses a large number of real images of hammer. Indicating incomplete detection.

However, in spite of balanced performance of the model to detect hammer, it struggles to correctly identify and detect instances of this class. While achieving a moderate precision the model tries hard to minimize false positive predictions. Even though the low recall highlights the model's capability to recognize and detect hammers within images.

The YOLOv7 model's performance for hammers highlights challenges in efficient object detection. The model shows that the precision-recall, data quality, model design and optimization strategies is vital to improve overall models' efficacy in detecting hammers and other objects in diverse image datasets.

6. Handgun:

The YOLOv7 model's performance for the class "Handguns" gives a precision score of 0.461 which indicates that this model successfully detects the images of this class object 46.1% of the time. This data indicates a moderate rate of false positives. On the other hand, the low recall score of 0.342 suggests that the model detects only 34.2% of all true instances of handguns. This indicates that the model misses a Significant portion of handguns. This indicates incomplete detection.

Although the model shows a balanced performance to detect handguns it still struggles to precisely identify the instances of this class. While achieving a moderate precision the model shows great effort to minimize false positives.

On the contrary a low recall rate highlights the model's capability to recognize and detect handguns.

The YOLOv7 model's performance for Handguns encounters difficulty in efficient object detection. The model shows that the precision-recall, data quality, model design and optimization strategies is essential to improve overall models' efficacy in detecting Handguns and other objects in diverse image datasets.

7. Knife:

The YOLOv7 model's performance for the class "Knife" gives a precision score of 0.456 which indicates that this model successfully detects the images of this class object 45.6% of the time. This data indicates a moderate rate of false positives. On the other hand, the low recall score of 0.296 suggests that the model detects only 29.6% of all true instances of knives. This indicates that the model misses a Significant portion of Knife. This indicates incomplete detection.

However, in spite of balanced performance of the model to detect Knife, it struggles to correctly identify and detect instances of this class. While achieving a moderate precision the model tries hard to minimize false positive predictions. Even though the low recall highlights the model's capability to recognize and detect Knife within images.

The YOLOv7 model's performance for Knife highlights challenges in efficient object detection. The model shows that the precision-recall, data quality, model design and optimization strategies is vital to improve overall models' efficacy in detecting Knife and other objects in diverse image datasets.

8. LeonardoCaprio:

The YOLOv7 model's performance concerning the "LeonardoCaprio" class displays the precision score as 0.367 which indicates that this model successfully detects the images of this class object 36.7% of the time. This indicates a moderate rate of false positive predictions where "LeonardoCaprio" is correctly detected.

On the other hand, the recall score for Leonardo DiCaprio is 0.724 which indicates that this model successfully detects the images of this class object 72.4% of the time, for all true instances of dataset. This high recall value indicates the efficacy of the model to identify instances of Leonardo DiCaprio. This moderate precision and high recall performance indicate minimal false positives. Overall the YOLOv7 model's performance for Leonardo DiCaprio effectively detects specific individuals within Image dataset. By achieving a certain balance, we see the model has achieved certain efficacy in detection of the tasks.

9. MarilynMonroe :

In detection of the class "MarilynMonroe" using the YOLOv7 model indicates that the precision score is at 0.75, which indicates that this model successfully detects the images of this class object 75% of the time. Suggesting a low false positive rate where the Marilyn Monroe is identified incorrectly.

On the other hand, the recall rate of Marilyn Monroe is rather impressive standing at 0.722. This high recall rate suggests that the model has high efficacy rate in detecting all true instances of Marilyn Monroe. This combination of the precision with impressive recall indicates the efficacy and competency of the YOLOv7 model. The high precision gives high accuracy and the minimizes the rate of false positives. The YOLOv7 model shows high accuracy in detecting all instances of Marylin Monroe contributing in overall proficiency in object detection within the dataset.

10. WillSmith:

The YOLOv7 model's performance for the class "WillSmith" gives a precision score of 0.62. This data indicates a moderate rate of false positives. On the other hand, the low recall score of 0.429 suggests that the model detects only 42.9% of all true instances of WillSmith. This indicates that the model misses a Significant portion of WillSmith. This indicates incomplete detection.

However, in spite of moderate precision of the model to detect WillSmith, it struggles to correctly identify and detect instances of this class. While achieving a moderate precision the model tries hard to minimize false positive

predictions. Even though the low recall highlights the model's capability to recognize and detect WillSmith within images.

The model suggests a room for improvement in recall, while achieving a fairly high precision is necessary, a higher recall would confirm more inclusive detection of Will Smith instances within the dataset.

The YOLOv7 model's performance for Will Smith highlights its ability to detect individuals with moderate precisions but also indicates limitations in its ability to inclusively capture all instances of this class. By improving the model's recall, we can work on improving the overall performances and efficacy of the model.

Overall Interpretation:

The results show that the YOLOv7 model shows a mixed performance across all classes some showing strong precision and recall, while others showing weak performance and limitation. Classes like "MarilynMonre" and "BillGates" display high precision and recall. Signifying accurate detection of all occurrences with in the dataset. The classes like "Axe" and bottle" displays weak performance displaying lower precision and recall. Signifying incorrect detection of all occurrences with in the dataset.

The mean average precision (MAP) values at IoU thresholds of 0.5 and across thresholds from 0.5 to 0.95 ranging from 0.0407 to 0.012 shows weaker performance moreover in rigorous IoU. These low MAP values highlight the model's restrictions in accurately localizing and detecting objects across various classes. The MAP score highlights the model's inclusive efficacy in object detection.

Overall, while the YOLOv7 model exhibits moderate efficiency across diverse classes. To improve its overall performance, targeted improvements in areas such as data quality, model architecture, and training strategies are vital. By addressing these factors and iteratively refining model, its efficacy, accuracy and reliability in object detection can be greatly enhanced, leading to more effective real-world usability.

The results of the YOLOv7 object detection are displayed in figure 6.2. The model has comparatively performed better than YOLOv5. It is able to detect Bill Gates but has poor accuracy with false positive results. It is able to detect Handgun, Knife and

other objects but still it is not sufficient enough for our research because of low accuracy.

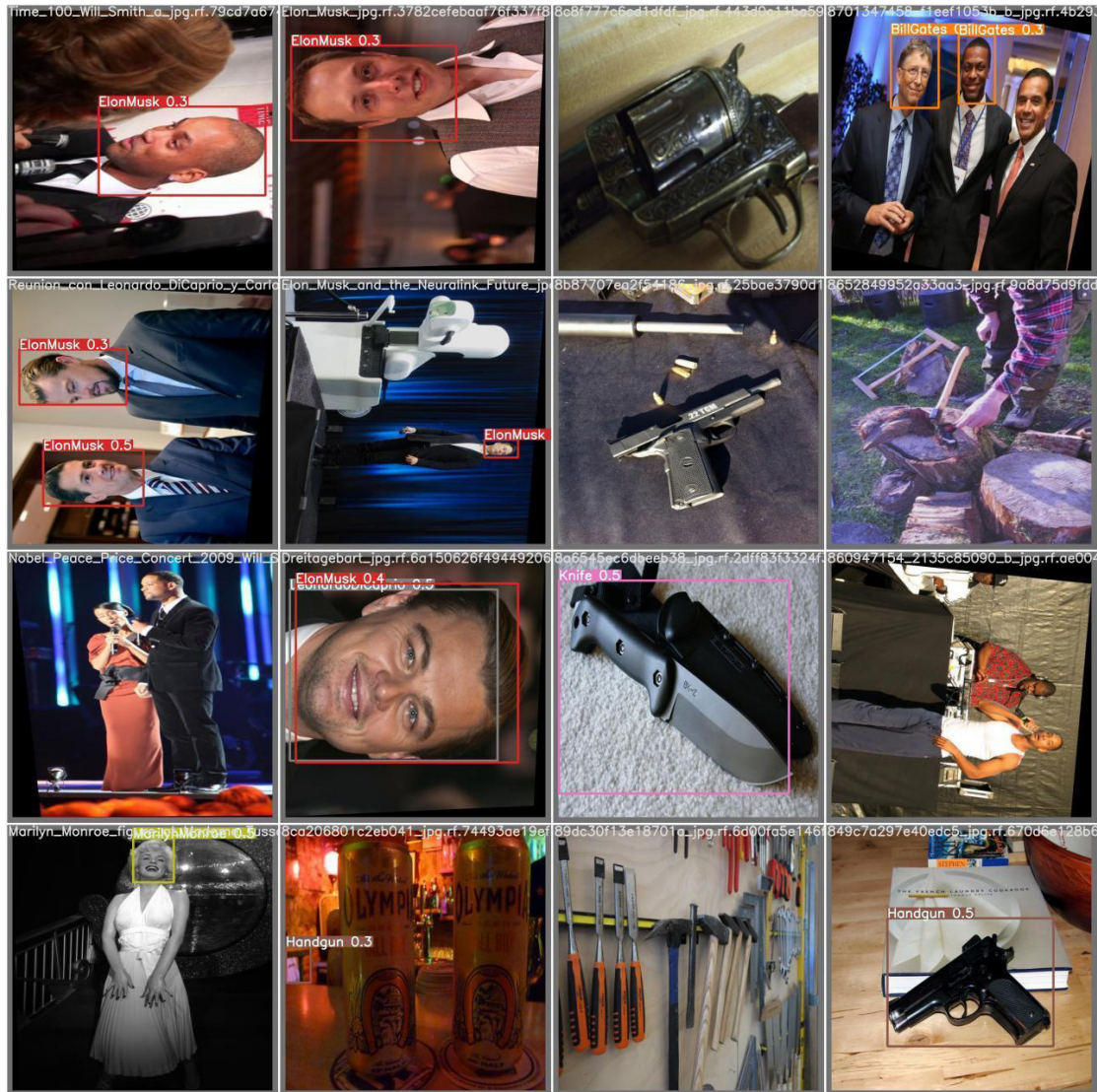


Fig. 6.2 : Result of YOLOv7 model on Image

6.3 YOLOv8 RESULTS

The table 6.3 summarizes the performance metrics of the YOLOv8 object detection model across different classes and overall, for our dataset.

Table 6.3 : Result of YOLOV8 object detection model

CLASS	IMAGES	LABELS	P	R	MAP@0.5	MAP@0.5:0.95
all	310	480	0.829	0.75	0.8	0.596
Axe	310	29	0.622	0.586	0.568	0.451

CLASS	IMAGES	LABELS	P	R	MAP@0.5	MAP@0.5:0.95
BillGates	310	32	0.951	1	0.992	0.61
Bottle	310	179	0.731	0.257	0.379	0.234
ElonMusk	310	20	0.861	1	0.99	0.696
Hammer	310	26	0.71	0.538	0.626	0.548
Handgun	310	40	0.901	0.683	0.804	0.613
Knife	310	54	0.586	0.629	0.674	0.56
LeonardoCaprio	310	29	1	0.94	0.992	0.783
MarilynMonroe	310	29	0.979	0.966	0.993	0.698
WillSmith	310	42	0.952	0.905	0.98	0.765

Class-Specific Interpretation:

1. Axe:

There is significant improvement in YOLOv8 with a precision value of 0.622. The model predicts all instances of the axe correctly with a significant reduction in false positive predictions as compared to the previous versions of YOLO. This indicates higher rate of accuracy in detecting axe objects.

The recall value of 0.586 increases the efficacy of the model to detect accurately 58.6% of all true instances of axes within the dataset. There is still some possibility for improvement in order to achieve a higher recall rate. The model indicates a significant transformation in enhanced recall and successful object detection. The Mean Average Precision (MAP) scores at IoU thresholds of 0.5 (0.568) and across thresholds from 0.5 to 0.95 (0.451) indicates enhanced performance for the "Axe" class making the model reliable in detecting axe across various images.

Overall, the YOLOv8 model's improved precision and recall metrics for the "Axe" class suggest higher accuracy and with minimal false positive predictions. YOLOv8 gives improved detection coverage compared to previous versions. More optimization and fine-tuning could possibly lead to higher efficacy, but the current results indicate significant development in object detection for this class.

2. BillGates:

Having a precision value of 0.951 this YOLOv8 model shows extra ordinary development. There is extraordinary development in YOLOv8 with the model predicts all occurrences of the Bill Gates correctly with a minimal false positive prediction as compared to the previous versions of YOLO. This indicates notable accuracy in detecting Bill Gates in images correctly.

The perfect recall of 1.0 indicates that model detects all instances correctly without missing on any true positives. This highlights the model's efficacy to detect Bill Gates across the dataset.

The Mean Average Precision (MAP) scores highlights the model's significant performance. With a MAP of 0.992 at IoU threshold 0.5 and 0.61 across thresholds from 0.5 to 0.95, the model achieves near-perfect identification. This model stands superior with astonishing precision and recall metrics for the "BillGates" class as compared to previous versions. The model highlights accuracy and dependability in detecting Bill Gates without losing precision and recall. This makes a treasured tool for the real-world people detection applications.

3. Bottle:

The YOLOv8 model detects bottle with a moderate precision and low recall with a precision value of 0.731 and successful identification 73.1% of the time. The results indicate some false positive and scope of improvement.

The recall value is 0.257, this low recall rate indicates that the model fails to detect many instances of the bottle, resulting in incomplete detection.

The MAP is 0.379 and MAP 0.5 to 0.95 is 0.234 which indicates poor efficacy. This displays the incompetency of the model to accurately detect bottles in images. Overall the YOLOv8 model displays decent precision for the "Bottle" class. However, the low recall suggests major challenges in effectively detecting the bottles.

Addressing factors such as data quality, model architecture, and training strategies could help improve the model's performance and enhance its ability to accurately identify bottles in diverse image datasets.

4. ElonMusk:

The YOLOv8 model performs exceptionally in identifying instances of Elon Musk in images as it achieves a high precision of 0.861 for the "ElonMusk" class with the successful identification 86.1% of the time, suggestive of rare false positives. The model also displays a perfect recall of 1.0 signifying successful detection of all true instances of ElonMusk Class. It often does not miss on the ElonMusk Image detection with a MAP of 0.99 at IoU threshold 0.5 and 0.696 across thresholds from 0.5 to 0.95.

The high level of performance of YOLOv8 model efficacy in precisely detecting people and its recall for the "ElonMusk" class, indicates its strength and dependability in identifying all instances of Elon Musk within varied image datasets. This is vital for various applications.

5. Hammer:

The YOLOv8 model shows dependability to detect all instances of hammer class with precision of 0.71. The model correctly detects approximately 71% of the time to achieve a precision of 0.71 for the "Hammer" class, indicating that when it predicts the presence of a hammer in an image, it is correct. The precision indicates low failure rate with reliability in correctly recognizing this class. However, the recall is 0.538, which shows that the model only identifies approximately 53.8% of all true instances of this class. This comparatively low recall rate shows that the model fails to identify hammers in a large number of actual instances which leads to incomplete detection.

The MAP scores of 0.626 at IoU threshold 0.5 and 0.548 across thresholds from 0.5 to 0.95, the model shows moderate accuracy.

This model's high precision and a low recall rate for hammers, indicates scope of improvement. Enhancements in data quality, model architecture, and training strategies may help in improving the model's performance for the "Hammer" class.

6. Handgun:

The yolov8 model shows proficient object detection for the class Handguns, with a Precision score of 0.901 and successful identification 90.1% of the

time. This high precision score indicates high proficiency and minimal false positive predictions.

The moderate recall score 0.683 suggest that the model successfully identifies approximately 68.3% of all actual instances of handguns indicating that the model may miss some objects. However, it demonstrates tremendous ability to detect Handguns across dataset.

The high MAP score 0.804, validates this model's efficacy and accuracy in identifying handguns. The MAP score and precision-recall balance, showcases the model's reliability and consistency. The satisfactory recall rate highlights minimal false positives.

7. Knife:

The precision score of 0.586 is improved compared to earlier versions, which indicates that it correctly identifies knife 58.6% of the time. This suggest that the model is more efficient in minimizing false positive predictions.

The Recall score of 0.629 indicates that the model successfully detects 62.9% of all true instances. This high recall signifies the high sensitivity of the model to detect knives while missing few true positives.

The moderate MAP score 0.674, validates this model's efficacy and accuracy in identifying knives. The MAP score and precision-recall balance, showcases the model's reliability and consistency. The high recall rate highlights minimal false positives.

8. LeonardoCaprio:

The YOLOv8 model achieves a flawless precision score of 1.0 for detecting the Leonardo DiCaprio class object. This model does not miss any instance of Leonardo DiCaprio. The model has no false positives which makes it extremely precise and consistent.

The model's high sensitivity is achieved with a recall score of 0.94. The model misses minimum instances while detecting 94% images of Leonardo DiCaprio correctly.

The excellent MAP score 0.992, and outstanding precision-recall, showcases the model's high performance and consistency. The model's excellent performance makes it flawless for real world usage.

9. MarilynMonroe:

The YOLOv8 model shows a high Precision score of 0.979 for this class MarilynMonroe by identifying it correctly 97.9% of the time. This high precision score indicates minimal false positive predictions.

With a recall score of 0.966, the Marilyn Monroe class item can be identified with high accuracy 96.6% of the time. This high recall indicates that the model has a high sensitivity to identify Marilyn Monroe with the fewest possible missed occurrences.

The exceptional MAP score 0.993 and outstanding precision-recall, makes the model perfect for real world usage.

10. WillSmith:

The YOLOv8 model shows proficient object detection for the class WillSmith, with a Precision score of 0.952 and successful identification 95.2% of the time. This high precision score indicates high proficiency and minimal false positive predictions.

The recall score of 0.905 shows a high accuracy of detecting Leonardo DiCaprio class object 90.5% of the time. This high recall signifies the high sensitivity of the model to detect Leonardo DiCaprio while minimizing the number of missed instances.

The high MAP score 0.98, validates this model's accuracy and reliability in identifying Marilyn Monroe. The high MAP score and outstanding precision-recall, showcases the model's reliability and consistency. The model's high performance makes it perfect for real world usage.

Overall Interpretation:

The overall interpretation of the YOLOv8 model's performance indicates its significant developments in object detection accuracy and efficiency compared to previous versions. The model demonstrates exceptional precision and recall values

across all classes, which indicates that it is able to accurately detect objects in the dataset.

The impressive MAP score further validates the model's reliability and consistency. However, some classes with lower recall values such as Bottle need improvements.

In conclusion, the YOLOv8 model's remarkable performance metrics, including high precision, recall, and MAP scores, highlight its effectiveness in accurately detecting and localizing objects of interest. While there are areas for improvement, the model's overall advancements signify a promising direction for the future of object detection technology.

The result of YOLOv8 model is displayed on the below image in figure 6.3. This image clearly shows that YOLOv8 performance is better than the yolov5 and yolov7. It is able to detect the objects correctly with more accuracy compared to previous models.

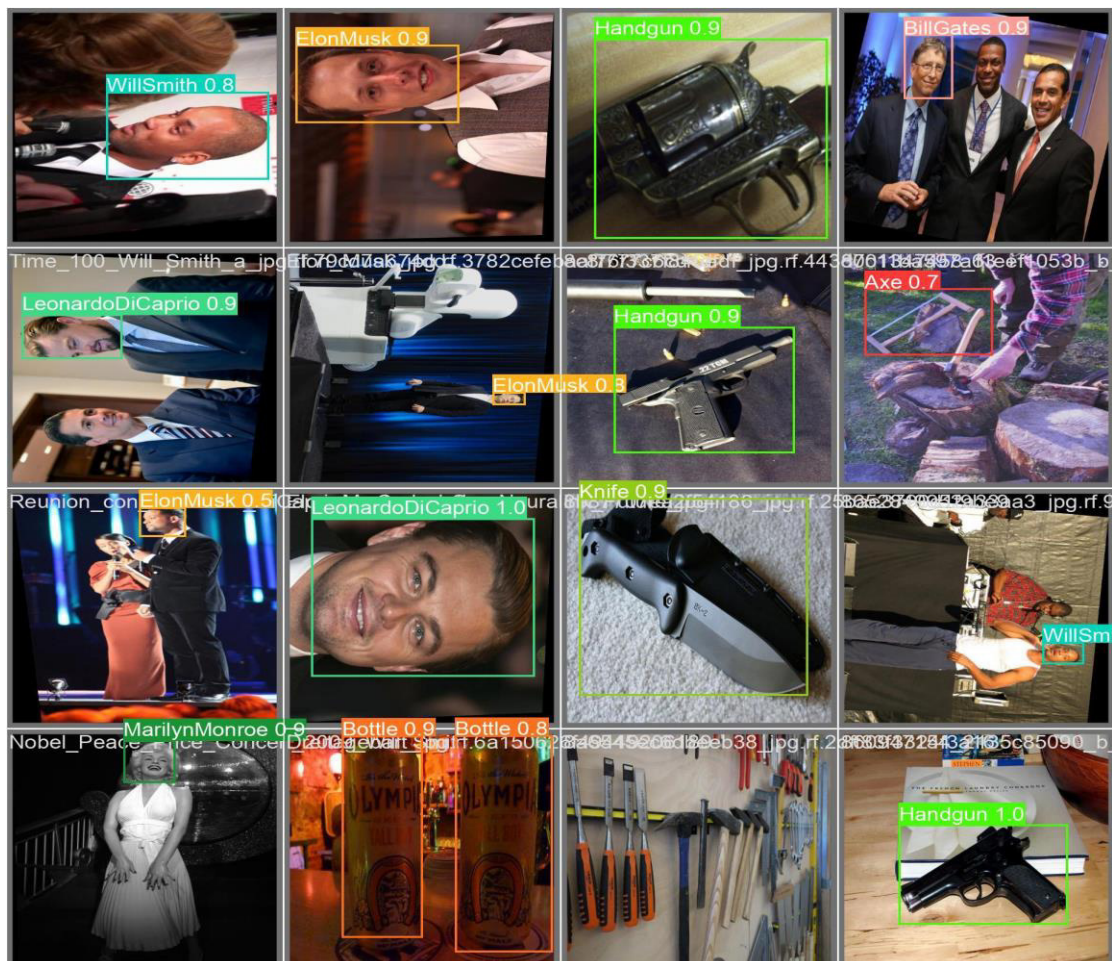


Fig. 6.3 : Result of YOLOv8 model on Image

6.4 COMPARATIVE ANALYSIS OF YOLOv5, YOLOv7 AND YOLOv8 RESULTS

Here is the comparative analysis of YOLOv5, YOLOv7, and YOLOv8 models trained on our custom dataset of 1550 images after pre-processing and augmentation.

The models are evaluated based on different evaluation metrics: Precision, Recall, F1 Score, mAP@0.5, and mAP@0.5:0.95. This comparison of 3 different versions of YOLO provides valuable insights and helps us take decision in real world scenarios. This class specific comparison also brings to light strengths and weaknesses of each versions

The YOLO (You Only Look Once) series have revolutionized object detection in the real-world scenario. Each succeeding version is aimed to improve upon previous versions.

Table 6.4 : Comparison of the various YOLO versions

	YOLOv5	YOLOv7	YOLOv8
Precision	0.37	0.435	0.829
Recall	0.246	0.474	0.75
F1 Score	0.296	0.454	0.788
mAP@0.5	0.0803	0.432	0.8
mAP@0.5:.95	0.0407	0.238	0.596
Completion time((in hours)	2.683	2.803	3.128
Weight(MB)	173.2	74.9	136.7
No. of Parameters	86278375	37245102	68133198

Based on Table 6.4, various metrics, characteristics and performance are compared and evaluated of YOLOv5, YOLOv7, and YOLOv8 models. Amongst the three versions, YOLOv8 is most advanced with its performance as compared to other versions. High Precision and minimal false positives make this version stand out. In contrast, YOLOv5 shows worst performance with its low precision scores. YOLOv7 gives moderate performance which is better than YOLOv5 but not proficient than YOLOv8. YOLOv8 demonstrates highest recall, F1 Score making it best model for capturing more true positives. This makes the model more dependable.

The mAP score of YOLOv8 outperforms both YOLOv5 and YOLOv7 enabling the model to detect objects with higher precision at a lower IoU threshold.

However, YOLOv8 has longest completion time than other versions which is indicative of more complex computations. YOLOv7 has the smallest model size 74.9MB preceding by YOLOv8 i.e. 136.7MB. The largest model size is of 173.2MB which is of YOLOv5.

The YOLOv7 model achieves high efficacy with a more compact model architecture. In terms of number of parameters, YOLOv8 falls between the two.

Overall YOLOv8 outperforms the other two models in terms of all the evaluation metrics.

Table 6.5 : mAP50 Performance of Individual class using YOLOv5, YOLOv7 and YOLOv8 models.

CLASS	YOLOV5L	YOLOV7L	YOLOV8L
all	0.0803	0.432	0.8
Axe	0.0261	0.175	0.568
BillGates	0.113	0.576	0.992
Bottle	0.0124	0.144	0.379
ElonMusk	0.192	0.712	0.99
Hammer	0.00839	0.158	0.626
Handgun	0.0252	0.325	0.804
Knife	0.0191	0.296	0.674
LeonardoCaprio	0.167	0.523	0.992
MarilynMonroe	0.0424	0.826	0.993
WillSmith	0.198	0.588	0.98

The Mean Average Precision (mAP) at IoU threshold 0.5 (mAP@0.5) provides a complete measure of the accuracy and localization capabilities of the object detection model. The mAP@0.5 scores of YOLOv5L, YOLOv7L, and YOLOv8L are compared in Table 6.5 to evaluate the performance of each model version.

The mAP@0.5 score of YOLOv5L suggest that the model faces challenges in localizing and detecting objects across classes at specified IoU threshold. However,

YOLOv5L gives a higher mAP@0.5 scores for "BillGates" and "MarilynMonroe" class, as compared to the other classes, indicative of better performance.

As compared to YOLOv5L the YOLOv7L shows significant improvements, across all classes, with an mAP@0.5 of 0.432. classes like "BillGates", "ElonMusk" and "MarilynMonroe" exhibit mAP@0.5 scores indicating of enhanced performance. This brings to light the development of YOLOv7L's over YOLOv5L.

The highest mAP@0.5 score of 0.8 is achieved by YOLOv8L. This signifies substantial developments in terms of detection, accuracy and localization as compared to the previous versions. There is a consistent high map score across all classes, whereas some classes like "BillGates," "ElonMusk," and "MarilynMonroe" achieve exceptional mAP@0.5 scores. This validates the model's ability to accurately detect objects with very few missed instances.

In terms of mAP@0.5 score each model starting from YOLOv5L shows improvement but YOLOv8L outperforms all models across classes. The consistent high mAP@0.5 scores across various classes in YOLOv8L makes it a robust and preferred choice for applications demanding high accuracy.

Table 6.6 : mAP50-95: Performance of Individual class using YOLOv5, YOLOv7 and YOLOv8 models.

CLASS	YOLOV5L	YOLOV7L	YOLOV8L
all	0.0407	0.238	0.596
Axe	0.012	0.0769	0.451
BillGates	0.0531	0.319	0.61
Bottle	0.00273	0.0503	0.234
ElonMusk	0.118	0.425	0.696
Hammer	0.00252	0.0637	0.548
Handgun	0.00768	0.121	0.613
Knife	0.0043	0.152	0.56
LeonardoCaprio	0.0938	0.357	0.783
MarilynMonroe	0.016	0.442	0.698
WillSmith	0.0968	0.371	0.765

The Mean Average Precision (mAP) at IoU threshold 0.5 to 0.95 (mAP50-95) provides a more rigorous evaluation of the object detection model's performance, considering a wider range of IoU thresholds. Comparing the mAP50-95 scores of YOLOv5L, YOLOv7L, and YOLOv8L in Table no. 6.6 reveals the efficiency of each versions in precisely localizing and detecting objects across different classes.

The lowest mAP50-95 score standing at 0.0407 achieved by YOLOv5L indicates the shortfalls of the model in accurately detecting objects with in a wider range of IoU thresholds. Despite some instabilities, classes like "BillGates" and "MarilynMonroe" exhibit higher mAP50-95 scores compared to others. Indicating relatively better performance across multiple IoU.

The YOLOv7L shows improvement with a mAP50-95 of 0.238 across all classes. This indicates better performance in detecting objects across different classes, when considering a wider range of IoU. Certain classes like "ElonMusk" and "WillSmith" demonstrate high mAP50-95 score, suggesting a superior performance detecting certain objects.

YOLOv8L demonstrates the highest mAP50-95 scores with the value of 0.596, signifying major improvements in detecting objects across various classes considering a wider range of IoU thresholds. The YOLOv8L with a consistent high mAP50-95, indicates major developments. YOLOv8L outperforms object detection with minimal missed instances and false positives.

The comparison of the three models from YOLOv5L to YOLOv8L highlights the improvements in comparison of mAP50-95 scores. The substantial developments in model architecture and training methodologies, lead to superior detection tasks.

Table 6.7 : Precision Performance of Individual class using YOLOv5, YOLOv7 and YOLOv8 models.

CLASS	YOLOV5L	YOLOV7L	YOLOV8L
all	0.37	0.435	0.829
Axe	1	0.307	0.622
BillGates	0.107	0.357	0.951
Bottle	0.0359	0.463	0.731

CLASS	YOLOV5L	YOLOV7L	YOLOV8L
ElonMusk	0.0708	0.287	0.861
Hammer	1	0.287	0.71
Handgun	0.132	0.461	0.901
Knife	1	0.456	0.586
LeonardoCaprio	0.116	0.367	1
MarilynMonroe	0.0296	0.75	0.979
WillSmith	0.21	0.62	0.952

Precision measures the number of true positive predictions among all positive predictions made by the model. Comparing the precision values across different classes for YOLOv5L, YOLOv7L, and YOLOv8L shown in Table 6.7 provides insights into the models' accuracy for each class.

YOLOv5L demonstrates a precision score of 0.37, indicating that the model predicts 37% of all instances accurately. In case of certain such as "Axe," "Hammer," "Knife," and "LeonardoCaprio" achieve perfect precision scores of 1. This means that the YOLOv5L model accurately detects all instances without any false positive. The other classes exhibit relatively lower precision values, indicating high amount of false positives.

The precision score for YOLOv7L has seen significant improvement as compared to the previous version. The precision score of 0.435 reflects improved accuracy. Certain classes such as "Axe," "Hammer," "Handgun," and "Knife" maintain high precision scores, indicating minimum false positives. On the other hand classes like "BillGates," "Bottle," "ElonMusk," "LeonardoCaprio," "MarilynMonroe," and "WillSmith" show significant improvement and higher accuracy, as compared to the precision score of YOLOv5L.

The highest and consistent precision among all predecessors is displayed by YOLOv8L with a value 0.829. The values indicate very few false positives and enhanced accuracy as compared to both the previous models. Classes like "BillGates," "ElonMusk," "LeonardoCaprio," "MarilynMonroe," and "WillSmith" have shown significant improvement in performance.

In all YOLOv8L out performs YOLOv7L, YOLOv5L in terms of precision. The consistent improvement in values high light the reliability and effectiveness in accurate predictions making it preferred model for object detection.

Table 6.8 : Recall Performance of Individual class using YOLOv5, YOLOv7 and YOLOv8 models.

CLASS	YOLOV5L	YOLOV7L	YOLOV8L
all	0.246	0.474	0.75
Axe	0	0.0345	0.586
BillGates	0.469	0.831	1
Bottle	0.00559	0.145	0.257
ElonMusk	0.7	0.95	1
Hammer	0	0.269	0.538
Handgun	0.05	0.342	0.683
Knife	0	0.296	0.629
LeonardoCaprio	0.724	0.724	0.94
MarilynMonroe	0.0345	0.722	0.966
WillSmith	0.476	0.429	0.905

Recall measures the proportion of true positive instances that the model correctly identifies among all actual positive instances in the dataset.

From Table 6.8 we can understand that, YOLOv5L has an overall recall of 0.246, giving it an accuracy of 24.6%. This model shows differed recall values across different classes. Certain classes like "BillGates," "ElonMusk," and "MarilynMonroe" achieve relatively high recall values. The high recall value is directly propionate to the efficiency and true positive instances. However, indicating the model effectively captures most true positive instances for these classes. On the other hand, classes such as "Axe," "Hammer," "Handgun," and "Knife" exhibit very low or zero recall values. YOLOv5L misses a significant portion of positive instances for these categories.

In Table 6.8, YOLOv7L shows advancement in overall recall compared to the previous version, with a value of 0.474, indicating better performance in detecting true positive instances. Certain classes like "BillGates," "ElonMusk," "MarilynMonroe," and "WillSmith" achieve relatively high recall values and showing

true positive instance. Also, the remainder of the classes showed significant improvement.

The highest recall is seen with the YOLOv8L with a value of 0.75, indicating superior performance detecting all true positive instances across all classes as shown in Table 6.8. This version gives a higher recall value consistently as compared to the previous versions. The classes like "BillGates," "ElonMusk," "MarilynMonroe," and "WillSmith" show enhanced ability of the model to capture most true positive instances.

YOLOv8L out performs the previous versions in terms of precision and object detection. While the YOLOv7L shows improvements in recall compared to the other two, YOLOv8L still make it the most a advanced and accurate for object detection.

Table 6.9 : F1 Score Performance of Individual class using YOLOv5, YOLOv7 and YOLOv8 models.

CLASS	YOLOV5L	YOLOV7L	YOLOV8L
all	0.2955	0.4537	0.7875
Axe	0.0000	0.0620	0.6035
BillGates	0.1742	0.4994	0.9749
Bottle	0.0097	0.2208	0.3803
ElonMusk	0.1286	0.4408	0.9253
Hammer	0.0000	0.2777	0.6121
Handgun	0.0725	0.3927	0.7770
Knife	0.0000	0.3590	0.6067
LeonardoCaprio	0.2000	0.4871	0.9691
MarilynMonroe	0.0319	0.7357	0.9725
WillSmith	0.2914	0.5071	0.9279

The F1 score combines both precision and recall into a single value, providing a balanced insight of a model's performance.

YOLOv5L achieves an overall F1 score of 0.2955, indicates a moderate balance between precision and recall. The model displays varied F1 score for different classes. The classes like "Axe," "Bottle," "Hammer," "Knife," and "LeonardoCaprio" exhibit

zero or significantly low F1 scores as shown in Table 6.9. This indicates limitation in the ability of the model to maintain a balance between recall and precision.

When it comes to YOLOv7L we see an overall improvement in the F1 score in Table 6.9, with a value of 0.4537, highlighting an enhanced performance in maintaining a balance between precision and recall. Just as YOLOv5L, certain classes like "BillGates" and "WillSmith" achieve improved F1 scores. On the other hand, classes such as "Axe," "Bottle," "Hammer," "Handgun," "Knife," "LeonardoCaprio," and "MarilynMonroe" display improved F1 scores.

Coming to the latest YOLOv8L we witness highest overall F1 score as compared to all previous versions. The value of 0.7875 indicates superior performance of the model in balancing precision and recall. YOLOv8L consistently achieves higher F1 scores across classes. Remarkably, classes such as "BillGates," "ElonMusk," "MarilynMonroe," and "WillSmith" show substantial improvements in F1 scores compared to previous versions as shown in Table 6.9.

The consistent development in improvements in F1 scores across various classes for YOLOv8L emphasize its dependability and effectiveness in achieving a balanced combination of precision and recall. Making this the most dependable and advanced model for accurate object detection tasks.

Precision Confidence Curve

A Precision-Confidence Curve is a graphical representation showing relationship amongst precision of the model's predictions and confidence scores.

- **Precision:** Precision measures the proportion of true positive detections (correctly identified objects) out of all positive detections (both true positives and false positives).
- **Confidence Score:** This is the score assigned by the model to indicate the likelihood that a predicted bounding box contains an object of interest. Higher confidence scores indicate greater certainty in the prediction.

Precision Confidence Curve of Yolov5, Yolov7 and Yolov8 is shown in figure 6.4, 6.5, and 6.6 respectively

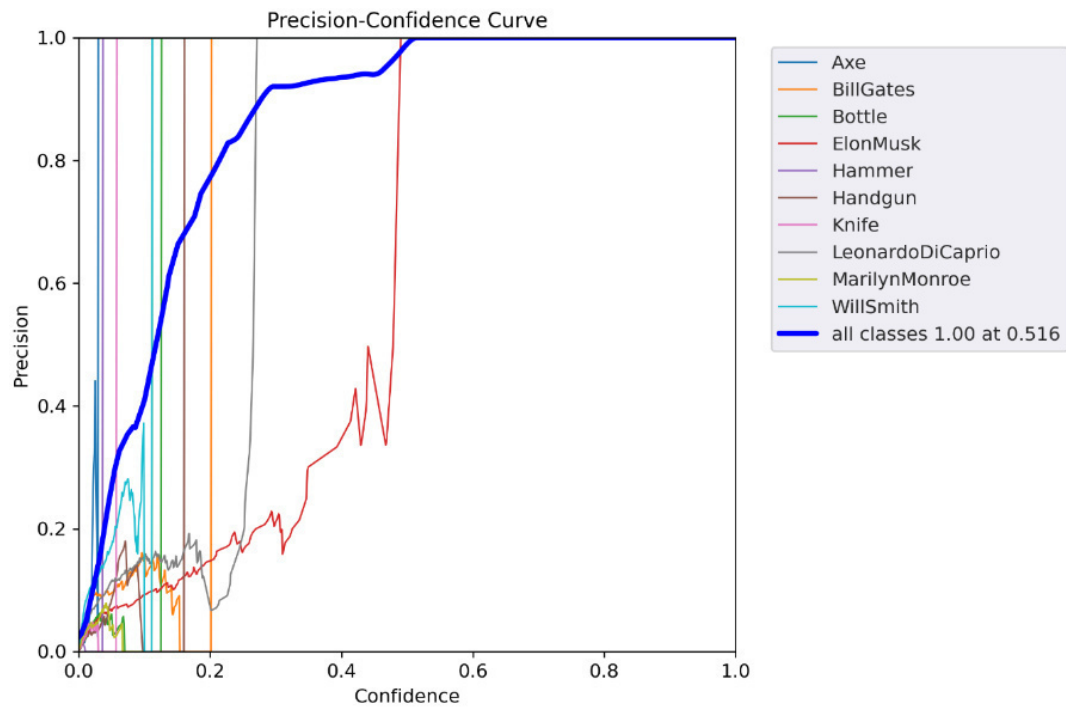


Fig. 6.4 : Precision -Confidence Curve of YOLOv5

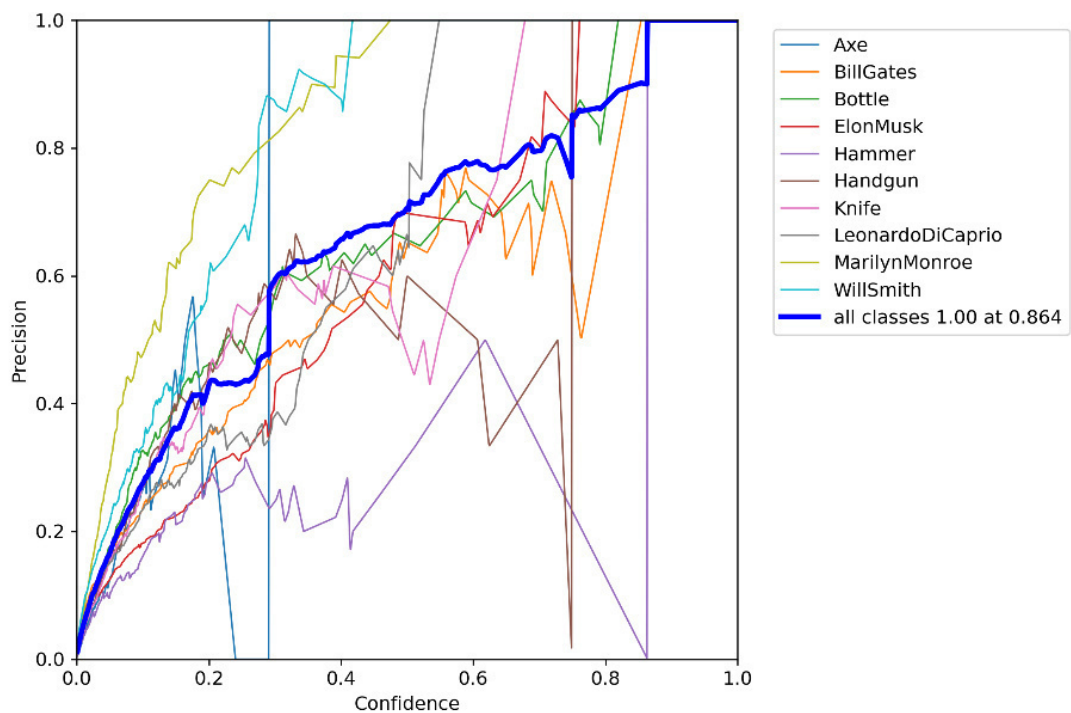


Fig. 6.5 : Precision -Confidence Curve of YOLOv7

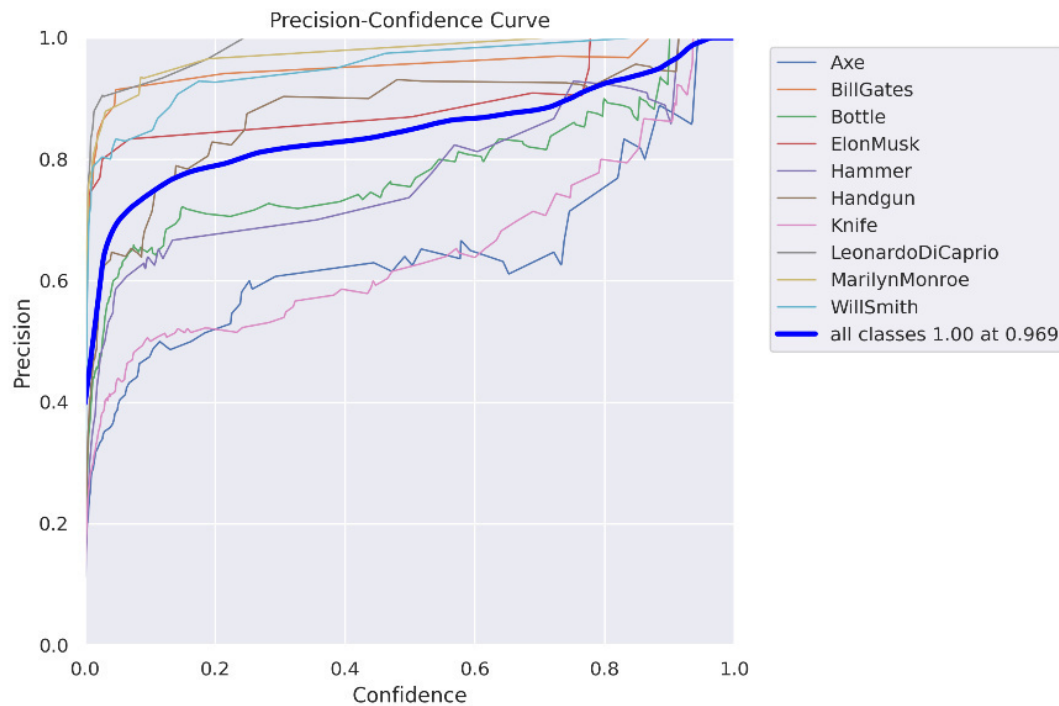


Fig. 6.6 : Precision -Confidence Curve of YOLOv8

When analyzing the precision-confidence curves for YOLOv8, YOLOv7, and YOLOv5, considering that all models achieve a precision of 1.00 at different confidence thresholds, the interpretation is as follows:

Precision-Confidence Curve Overview

1. Precision:

- Precision measures the accuracy of positive predictions made by the model (i.e., the proportion of true positives among all positive predictions).
- A precision of 1.00 means there are no false positives within the predictions classified as positive above a certain confidence threshold.

2. Confidence Threshold:

- The confidence threshold is the minimum score at which the model considers a prediction to be positive.
- Higher confidence thresholds mean the model is more selective about which predictions it classifies as positive.

YOLOv8, YOLOv7, and YOLOv5 Precision-Confidence Analysis

YOLOv8 achieves perfect precision (1.00) at a notably high confidence threshold of 0.969 indicating a confident model ensuring minimum false positive above this threshold as shown in Figure 6.6.

Figure 6.5 shows that the YOLOv7 achieves a perfect precision (1.00) with a slightly lower confidence score. Despite lower threshold YOLOv7 maintains high accuracy for positive predictions.

Figure 6.4 displays that the YOLOv5 with a lower confidence threshold of 0.516 with a precision (1.00). The YOLOv5 demonstrates significantly higher tolerance for positive detections while detecting a larger range of prediction confidence.

Model Reliability and Stringency:

In the processes of comparison of YOLOv8, YOLOv7, and YOLOv5, it is discovered that YOLOv8 displays higher threshold for perfect precision. This ensures predictions only when the model is confident. In contrast YOLOv7 operates at a moderate threshold, striking a balance between stringency and reliability, avoiding false positives within broader true positives.

Recall-Confidence Curve

Recall-Confidence Curve in YOLO Object Detection demonstrates how recall varies with different confidence thresholds for an object detection model. Recall measures the proportion of actual positives correctly identified by the model. At a confidence threshold of 0.000, every detected instance, regardless of confidence level, is counted as positive.

In this context, analyzing recall at a confidence threshold of 0.000 for YOLOv5, YOLOv7, and YOLOv8 reveals how well each model captures all possible positives without considering the confidence score. Recall-Confidence Curve of YOLOv5, YOLOv7 and YOLOv8 is shown in figure 6.7, 6.8, and 6.9 respectively

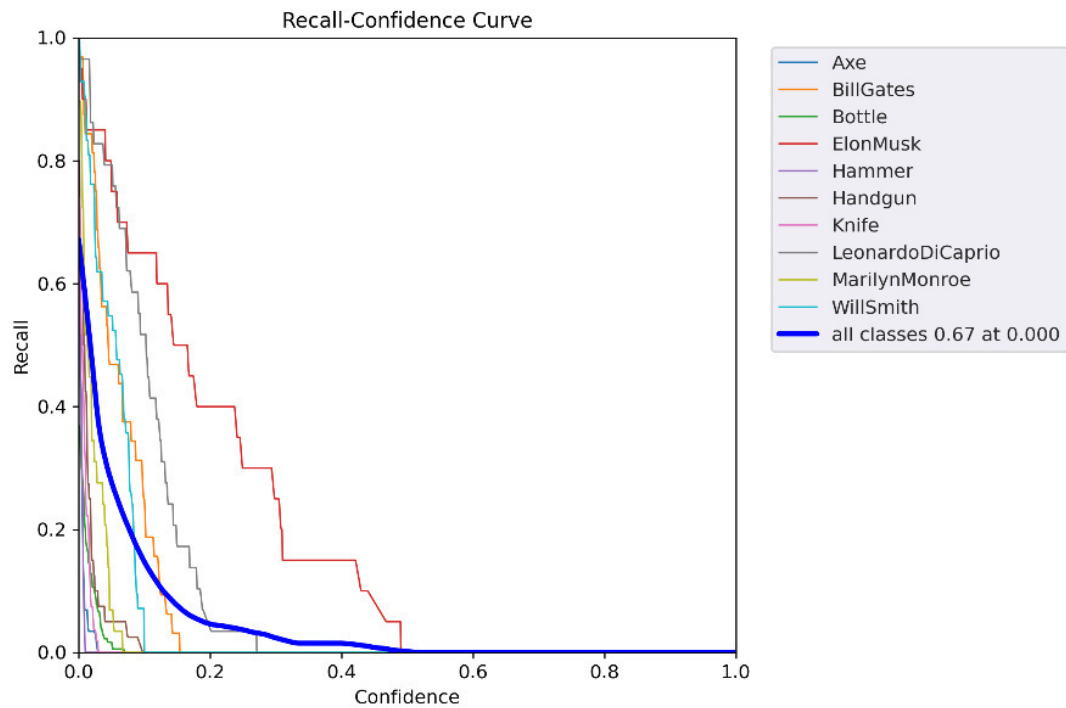


Fig. 6.7 : Recall -Confidence Curve of YOLOv5

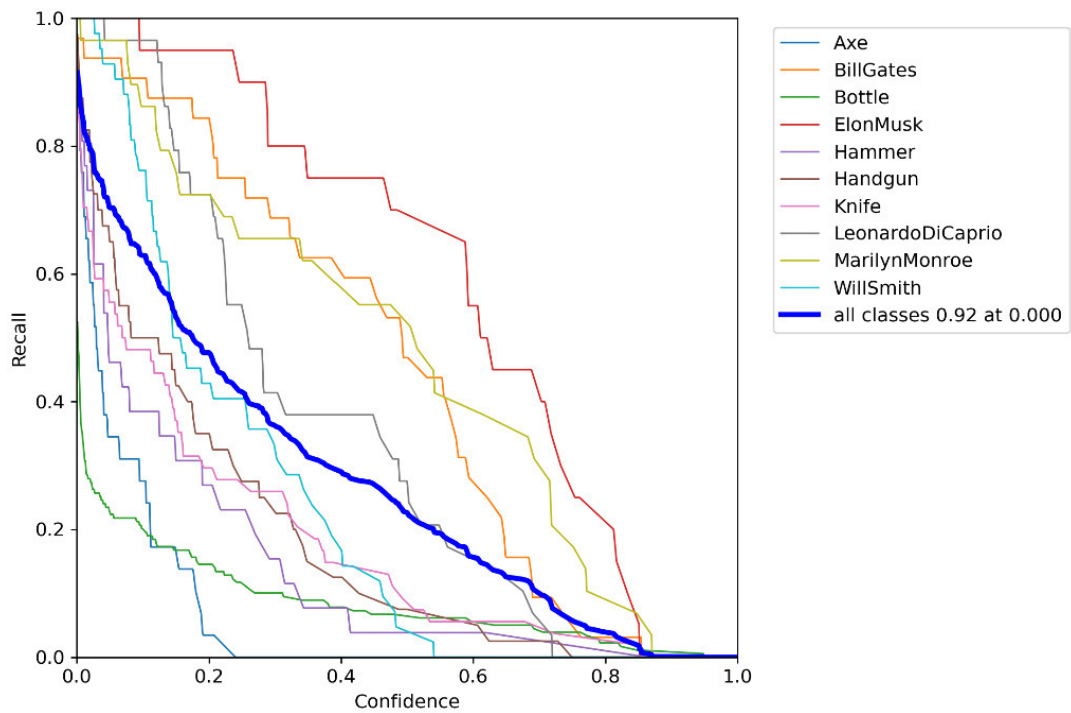


Fig. 6.8 : Recall -Confidence Curve of YOLOv7

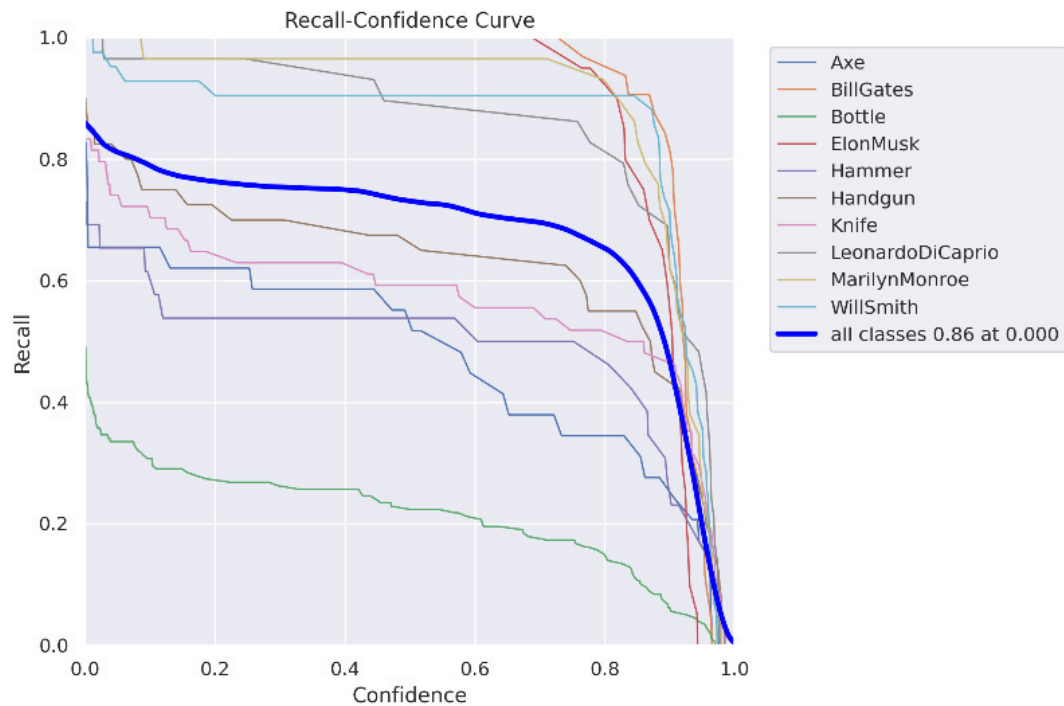


Fig. 6.9 : Recall -Confidence Curve of YOLOv8

In evaluating the recall performance of YOLOv5, YOLOv7, and YOLOv8 across varying confidence thresholds, distinct patterns emerge that influence their suitability for different applications. YOLOv7 stands out with the highest recall rate of 0.92 at a confidence threshold of 0.000 as shown in Figure 6.8, indicating its exceptional ability to capture 92% of all true positive instances without filtering based on confidence. This makes YOLOv7 particularly well-suited for tasks where comprehensive detection of every possible positive instance is essential. Following closely, YOLOv8 achieves a recall of 0.86 under similar conditions as shown in Figure 6.9, showcasing strong performance in identifying a significant portion of true positives while potentially offering more selective detections at higher confidence levels. In contrast, Figure 6.7 shows that the YOLOv5 exhibits a lower recall of 0.67 at the lowest confidence threshold, suggesting it captures a moderate proportion of true positives compared to YOLOv7 and YOLOv8.

Precision-Recall Curve in YOLO Object Detection

The precision-recall curve is a critical tool in evaluating object detection models, illustrating the trade-off between precision (the proportion of true positive detections among all positive detections) and recall (the proportion of true positive detections

among all actual positives) for different confidence thresholds. In the context of YOLO models, these metrics reveal the performance of each version in accurately detecting and classifying objects. The Precision- Recall Curve of YOLOv5, YOLOv7 and YOLOv8 is shown in figure 6.10,6.11, and 6.12 respectively.

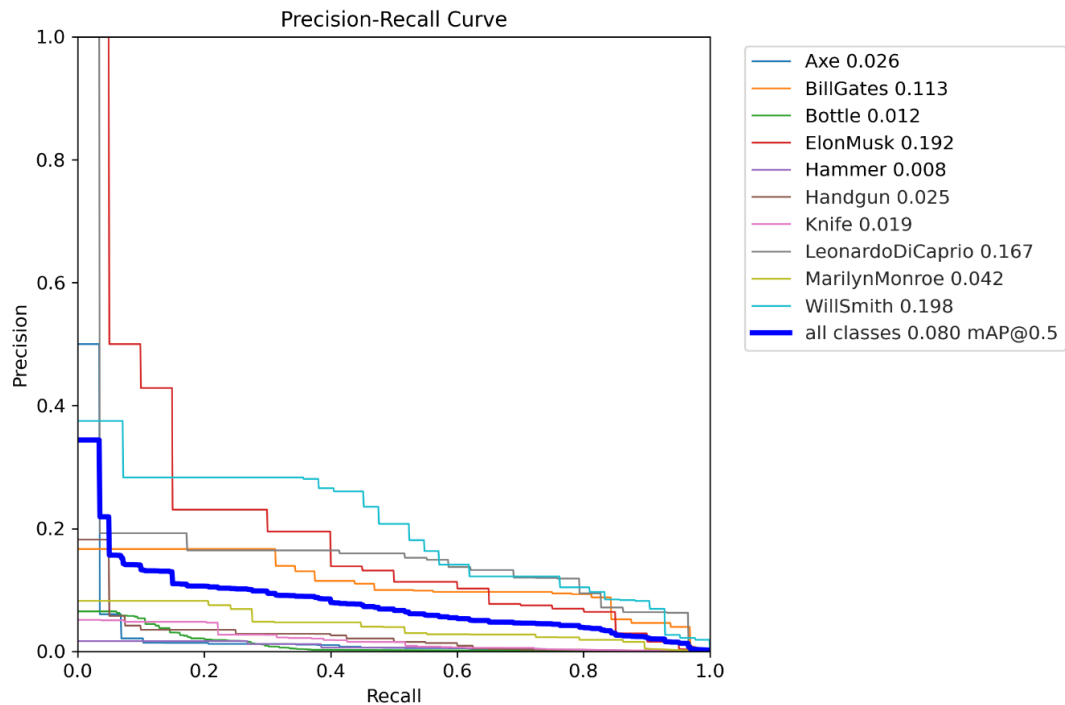


Fig. 6.10 : Precision- Recall Curve of YOLOv5

For YOLOv5, the precision-recall analysis reveals a mAP@0.5 (mean Average Precision at a 0.5 Intersection over Union threshold) of 0.080 as shown in figure 6.10. This low score indicates that YOLOv5 struggles significantly with both precision and recall, resulting in a high number of false positives and missed detections. Essentially, YOLOv5's object detection capabilities are limited, making it less reliable for applications requiring high accuracy.

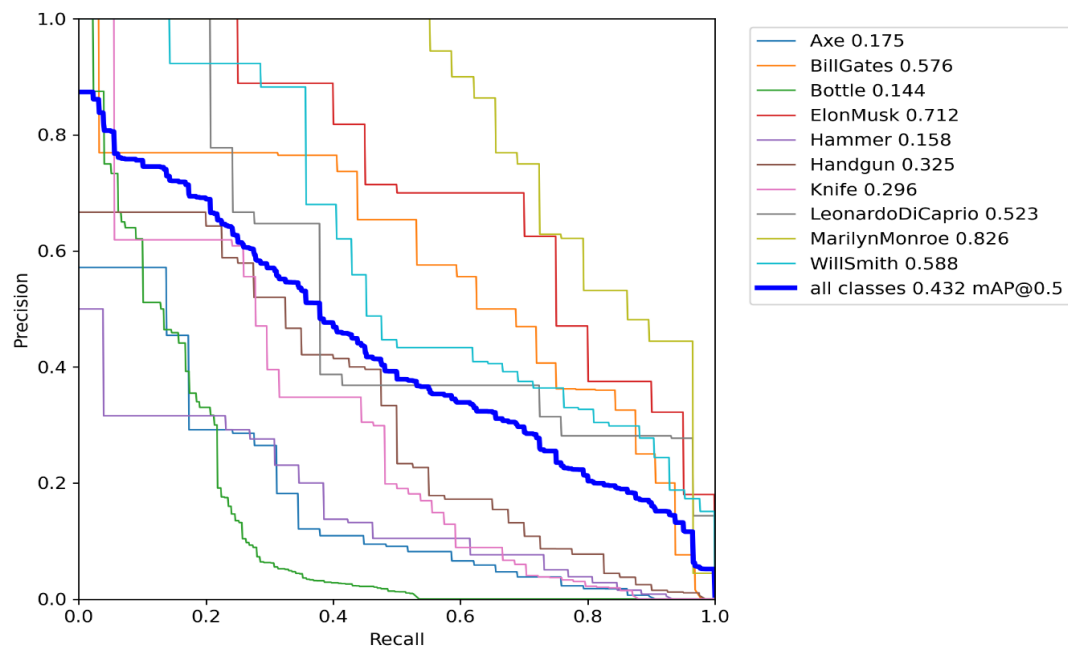


Fig. 6.11 : Precision- Recall Curve of YOLOv7

In contrast, YOLOv7 shows a marked improvement with a mAP@0.5 of 0.432 as shown in figure 6.11. This indicates a moderate level of performance where precision and recall are better balanced compared to YOLOv5. YOLOv7 can detect and classify objects more accurately, leading to fewer false positives and false negatives. This moderate performance suggests that YOLOv7 is suitable for applications where a reasonable level of accuracy is acceptable but not critical.

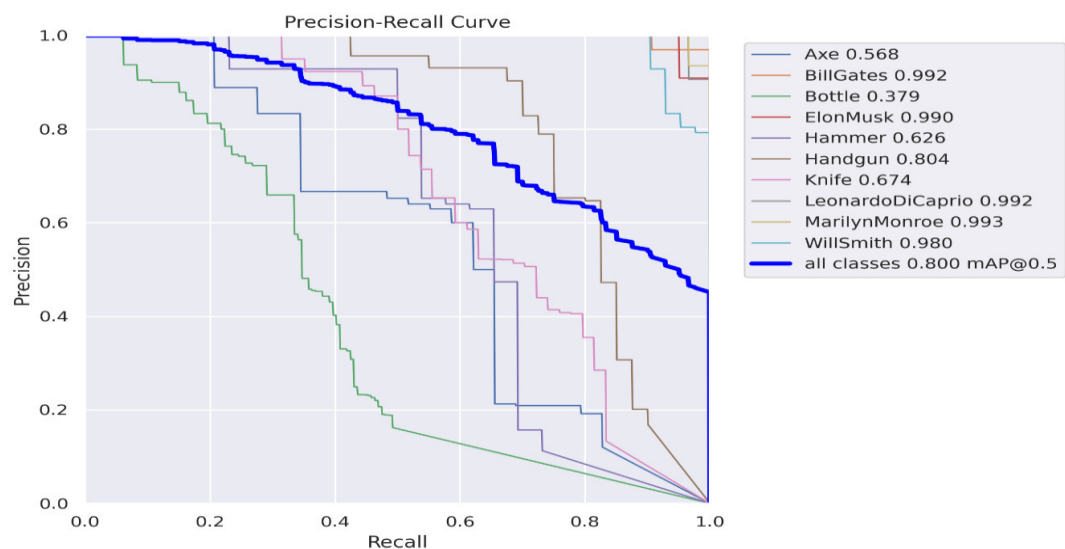


Fig. 6.12 : Precision- Recall Curve of YOLOv8

YOLOv8, however, outperforms both YOLOv5 and YOLOv7, with a high mAP@0.5 of 0.800 as shown in figure 6.12. This high score signifies excellent performance in terms of both precision and recall. YOLOv8 effectively detects and classifies objects with minimal errors, making it the most reliable model among the three. The high precision and recall indicate that YOLOv8 can accurately identify objects with few false positives and false negatives.

In summary, the precision-recall analysis clearly distinguishes the varying strengths of YOLOv5, YOLOv7, and YOLOv8. YOLOv5 shows the lowest performance with significant room for improvement, YOLOv7 offers a better balance and moderate performance, and YOLOv8 excels with the highest accuracy and reliability. These insights are crucial for selecting the appropriate model based on the specific needs of real-world applications.

F1 Confidence Curve

The F1 confidence curve is a vital evaluation tool in object detection models, combining both precision and recall into a single metric. The F1 score is the harmonic mean of precision and recall, providing a balance between the two. The confidence curve shows how the F1 score varies with different confidence thresholds, revealing the model's performance across a range of detection confidences. The F1 -Confidence Curve of Yolov5, Yolov7 and Yolov8 is shown in figure 6.13,6.14, and 6.15 respectively

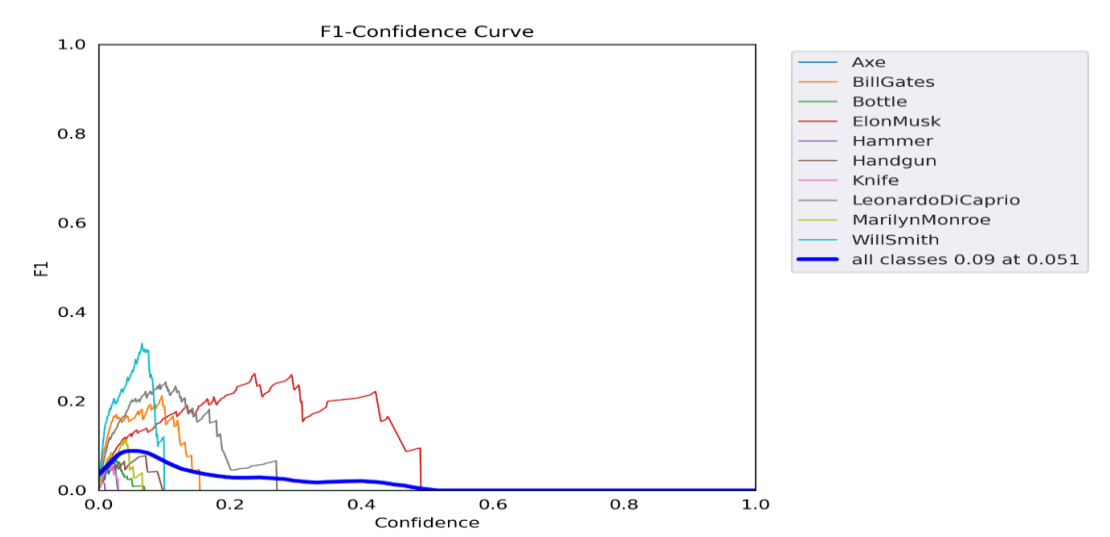


Fig. 6.13 : F1 Confidence Curve of YOLOV5

For YOLOv5, the F1 score at its best is 0.09 at a confidence threshold of 0.051 as shown in figure 6.13. This indicates that YOLOv5 performs poorly in balancing precision and recall, leading to a low F1 score. The low threshold at which this F1 score is achieved suggests that even at minimal confidence levels, the model struggles to identify and classify objects accurately, resulting in a high number of false positives and false negatives. This makes YOLOv5 less suitable for applications where accuracy is crucial.

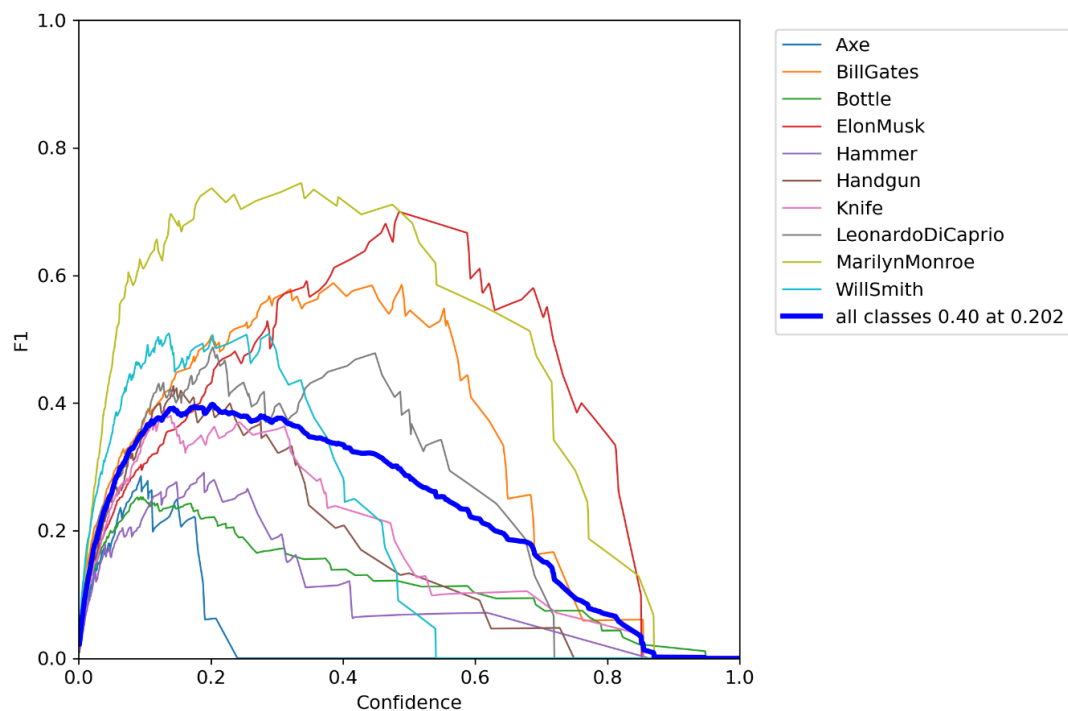


Fig. 6.14 : F1 Confidence Curve of YOLOv7

Figure 6.14 shows a significant improvement in YOLOv7, with an F1 score of 0.40 at a confidence threshold of 0.202. This higher F1 score reflects a better balance between precision and recall compared to YOLOv5. The model performs more reliably across various confidence levels, reducing the number of misclassifications. YOLOv7's performance indicates its suitability for applications where a moderate level of accuracy is acceptable and beneficial.

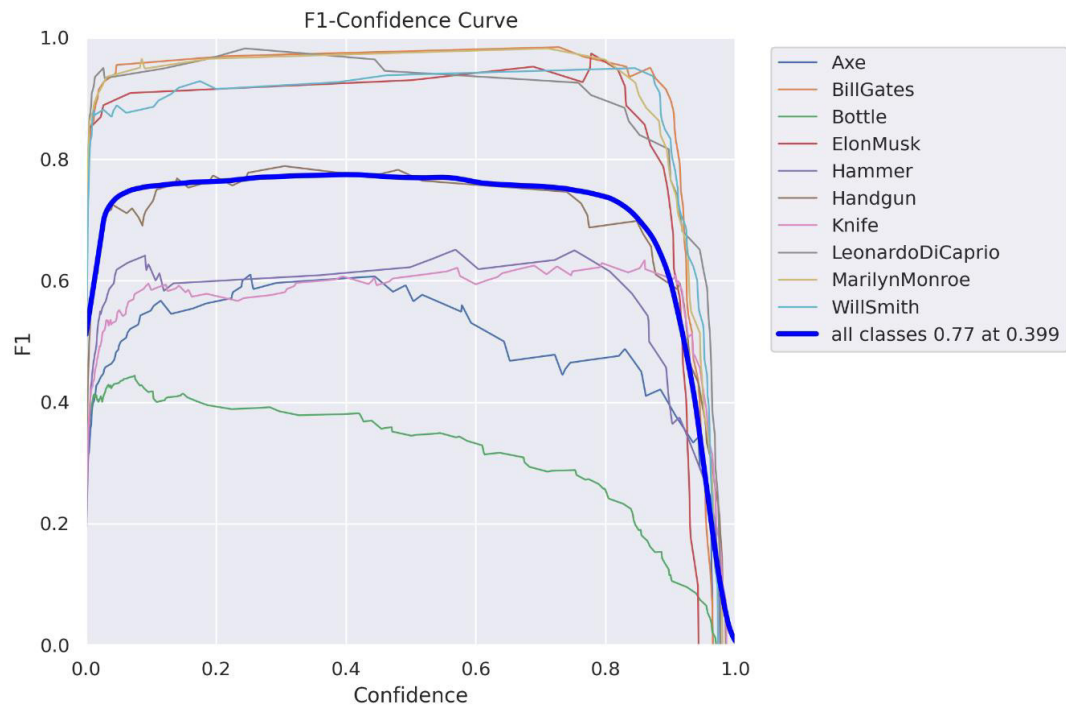


Fig. 6.15 : F1 Confidence Curve of YOLOv8

Through figure 6.15 we can understand that, YOLOv8 demonstrates the best performance, with an F1 score of 0.77 at a confidence threshold of 0.399. This high F1 score indicates that YOLOv8 excels in achieving a balance between precision and recall, making accurate detections with fewer errors. The higher confidence threshold shows that the model maintains its reliability even as the confidence level increases, making it ideal for applications requiring high accuracy and minimal false detections. The superior F1 score of YOLOv8 underscores its effectiveness in detecting and classifying objects accurately, making it the most reliable choice among the three models.

In summary, the F1 confidence curve analysis highlights the varying capabilities of YOLOv5, YOLOv7, and YOLOv8 in object detection. YOLOv5 shows the weakest performance, YOLOv7 offers moderate improvement, and YOLOv8 provides the best balance of precision and recall, making it the most accurate and reliable model for real-world applications demanding high accuracy and minimal errors.

CONFUSION MATRIX

A confusion matrix for multiple classes provides a detailed breakdown of prediction results. Each row represents the actual class, while each column represents the predicted class. The diagonal elements represent the true positives (correct predictions), and off-diagonal elements represent false positives (misclassifications).

The Confusion Matrix of Yolov5, Yolov7 and Yolov8 is shown in figure 6.16,6.17, and 6.18 respectively

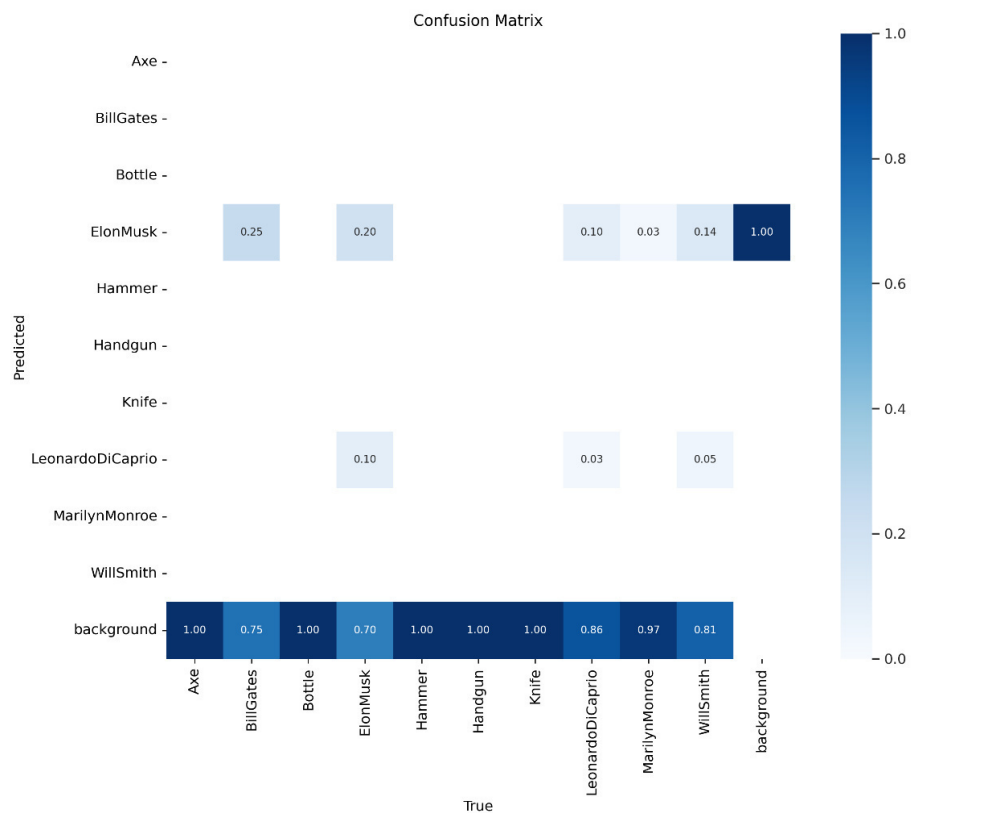


Fig. 6.16 : Confusion Matrix of YOLOv5



Fig. 6.17 : Confusion Matrix of YOLOv7

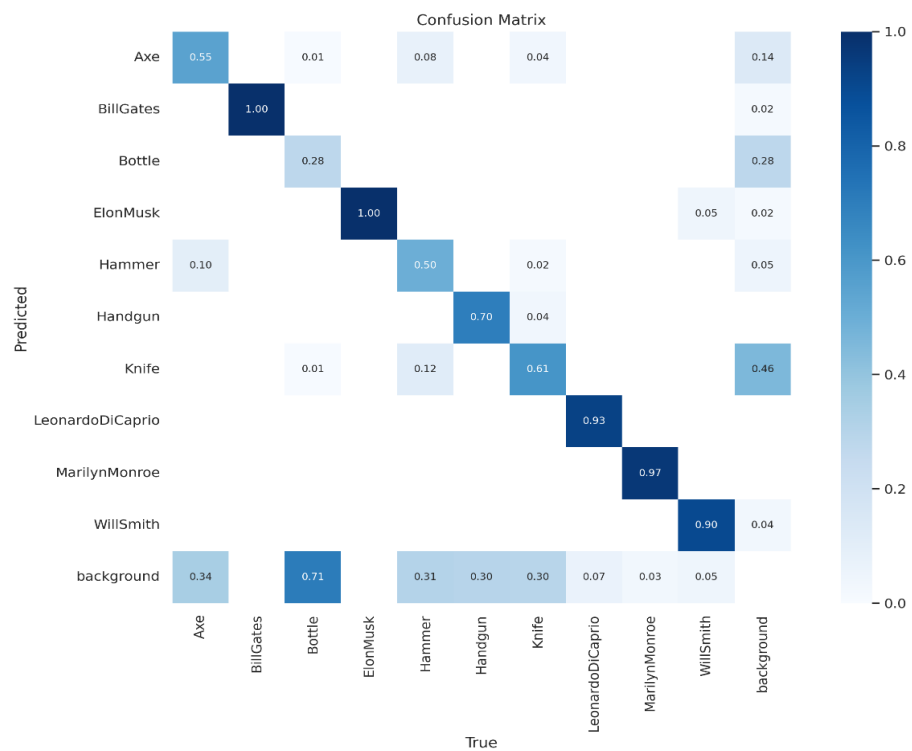


Fig. 6.18 : Confusion Matrix of YOLOv8

From the above figures of confusion matrices of yolov5, yolov7 and yolov8 we can interpret that yolov8 performs the best amongst other two. There are more diagonal elements in YOLOv8 confusion matrix compared to that of other two models.

Conclusion

The comparative analysis of YOLOv5, YOLOv7, and YOLOv8 reveals a clear progression in object detection capabilities, with YOLOv8 demonstrating the highest precision, recall, and MAP scores among the three. YOLOv5 shows moderate performance with balanced but subpar metrics, while YOLOv7 offers improvements over YOLOv5, especially in certain classes, yet falls short of YOLOv8's overall performance. YOLOv8 excels across most classes, though minor inconsistencies remain. The findings highlight YOLOv8 as the most effective version.

CHAPTER – VII

CONCLUSION AND FUTURE WORK



7.1 Summary of Findings

In this study, I conducted a comprehensive comparative analysis of three versions of the YOLO object detection algorithm: YOLOv5, YOLOv7, and YOLOv8. My analysis involved evaluating these models on a dataset of 1550 images across various performance metrics, including precision, recall, F1 score, and mean Average Precision (mAP) at different Intersection over Union (IoU) thresholds.

Table 7.1 : Overall Performance

Metric	YOLOv5	YOLOv7	YOLOv8
Precision	0.37	0.435	0.829
Recall	0.246	0.474	0.75
F1 Score	0.296	0.454	0.788
mAP@0.5	0.0803	0.432	0.8
mAP@0.5:0.95	0.0407	0.238	0.596

Precision: YOLOv8 demonstrates a substantial improvement in precision compared to YOLOv5 and YOLOv7, indicating its enhanced capability in correctly identifying true positives as shown in Table 7.1.

Recall: Table 7.2 displays, YOLOv8's higher recall reflects its proficiency in capturing a greater number of relevant instances, reducing the chances of missing important detections.

F1 Score: With the highest F1 score, YOLOv8 balances both precision and recall effectively, making it the most reliable model among the three.

mAP@0.5: The significant leap in mAP@0.5 for YOLOv8 underscores its superior accuracy in object detection tasks.

mAP@0.5:0.95: YOLOv8 excels in this more stringent metric, highlighting its consistent performance across varying IoU thresholds.

Class-Specific Performance

Our class-specific performance analysis further substantiates the advancements in YOLOv8 across a diverse set of categories.

Table 7.2 : Class-Specific Performance

Class	YOLOv5 mAP@0.5	YOLOv7 mAP@0.5	YOLOv8 mAP@0.5
Axe	0.0261	0.175	0.568
BillGates	0.113	0.576	0.992
Bottle	0.0124	0.144	0.379
ElonMusk	0.192	0.712	0.99
Hammer	0.00839	0.158	0.626
Handgun	0.0252	0.325	0.804
Knife	0.0191	0.296	0.674
LeonardoCaprio	0.167	0.523	0.992
MarilynMonroe	0.0424	0.826	0.993
WillSmith	0.198	0.588	0.98

Axe: YOLOv8 shows a marked improvement, increasing the mAP@0.5 from 0.0261 (YOLOv5) and 0.175 (YOLOv7) to 0.568 as shown in table 7.2

BillGates: Achieves near-perfect detection with an mAP@0.5 of 0.992.

Bottle: Table 7.2 shows Significant improvement in YOLOv8, with mAP@0.5 rising to 0.379.

ElonMusk: Near-perfect performance with an mAP@0.5 of 0.99 of YOLOv8 model.

Hammer: YOLOv8 demonstrates improved detection with an mAP@0.5 of 0.626.

Handgun: Substantial improvements in detection accuracy with YOLOv8.

Knife: Significant performance increase in YOLOv8.

LeonardoCaprio: High detection accuracy with an mAP@0.5 of 0.992.

MarilynMonroe: Near-perfect detection with an mAP@0.5 of 0.993.

WillSmith: High detection accuracy with an mAP@0.5 of 0.98.

Recall

Our recall analysis further highlights the advancements in YOLOv8.

Table 7.3 : Recall

Class	YOLOv5 Recall	YOLOv7 Recall	YOLOv8 Recall
Axe	0	0.0345	0.586
BillGates	0.469	0.831	1
Bottle	0.00559	0.145	0.257
ElonMusk	0.7	0.95	1
Hammer	0	0.269	0.538
Handgun	0.05	0.342	0.683
Knife	0	0.296	0.629
LeonardoCaprio	0.724	0.724	0.94
MarilynMonroe	0.0345	0.722	0.966
WillSmith	0.476	0.429	0.905

Axe: YOLOv8 shows remarkable improvements, achieving a recall of 0.586 as displayed in Table 7.3

BillGates: Table 7.3 shows Perfect recall in YOLOv8, indicating all instances were correctly identified.

Bottle: YOLOv8 improves recall significantly to 0.257 as displayed in Table 7.3.

ElonMusk: Perfect recall in YOLOv8, highlighting its reliability.

Hammer: Improved recall to 0.538 in YOLOv8.

Handgun: Significant improvements in recall with YOLOv8.

Knife: Enhanced recall to 0.629 in YOLOv8.

LeonardoCaprio: High recall of 0.94 in YOLOv8.

MarilynMonroe: Near-perfect recall of 0.966 in YOLOv8.

WillSmith: High recall of 0.905 in YOLOv8.

7.2 Contributions and Implications of the Study

Enhanced Object Detection

This study demonstrates how YOLOv8 significantly enhances object detection accuracy and speed. The improvements in precision and recall metrics indicate that YOLOv8 can reliably identify and classify objects with higher confidence and fewer

errors. This enhancement is particularly crucial for real-time applications where quick and accurate detections are necessary.

Facial Recognition Advancements

Incorporating facial recognition within the YOLO framework presents a novel approach to early victim identification in forensic investigations. The high accuracy of YOLOv8 in detecting and recognizing specific individuals (e.g., celebrities) suggests its potential utility in forensic applications, where accurate identification can lead to quicker and more effective investigations.

Resource Efficiency

The study highlights how newer models like YOLOv8 manage to balance high performance and computational efficiency. This balance makes YOLOv8 suitable for deployment in resource-constrained environments, such as mobile devices and edge computing scenarios, where processing power and memory are limited.

Forensic Applications

The research underscores the potential of advanced YOLO models in forensic science. The high accuracy and reliability of YOLOv8 in identifying objects and individuals can significantly aid forensic experts in analyzing crime scenes, identifying suspects, and gathering evidence. This application is particularly relevant in scenarios requiring high accuracy and quick turnaround times.

7.3 Future Research Directions in YOLO

Exploring Lightweight Models

Future work could focus on developing lightweight versions of YOLOv8 that retain high accuracy while being more efficient for deployment on edge devices. This direction involves optimizing the model architecture and reducing its computational requirements without compromising detection performance.

Integration with Other Technologies

Combining YOLOv8 with other AI technologies, such as natural language processing and context-aware computing, can enhance its application in more complex forensic scenarios. For instance, integrating YOLOv8 with a contextual analysis framework could provide deeper insights into the detected objects and their surroundings, improving the overall forensic analysis process.

Extended Datasets

Applying the models to larger and more diverse datasets can further validate and refine their performance. Expanding the dataset to include more varied and challenging images will help ensure that the models generalize well across different scenarios and improve their robustness and reliability.

Real-time Applications

Investigating the application of YOLOv8 in real-time forensic analysis, including live video feeds, is crucial to assess its practical utility and performance under operational conditions. Real-time applications require the model to process and analyze video frames swiftly, making it essential to optimize YOLOv8 for such tasks.

7.4 Concluding Remarks

The comparative analysis of YOLOv5, YOLOv7, and YOLOv8 on a dataset of 1550 images demonstrates clear advancements with each version. YOLOv8 significantly outperforms its predecessors across all key metrics, including precision, recall, and average precision (mAP), showcasing its superior capability in object detection and classification tasks. The study's contributions highlight the practical implications for enhanced object detection and facial recognition, especially in forensic applications. Future research directions suggest exploring lightweight models, integration with other AI technologies, validation on larger datasets, and real-time applications to further advance the field.

In conclusion, YOLOv8 represents a significant step forward in the field of object detection, offering substantial improvements over earlier versions. Its enhanced performance metrics and potential applications in real-time and forensic scenarios underscore its importance and utility in advancing both academic research and practical implementations in various domains

BIBLIOGRAPHY



BIBLIOGRAPHY

1. al, J. K. (July 2016). PIZZARO: Forensic analysis and restoration of image and video data. *Forensic Science International*, 264, 153-166. doi:<https://doi.org/10.1016/j.forsciint.2016.04.027>
2. Babenko, A., Slesarev, A., Chigorin, A., & Lempitsky, V. (2014). Neural Codes for Image Retrieval. *European Conference on Computer Vision*, Springer, vol 8689(Lecture Notes in Computer Science). doi:https://doi.org/10.1007/978-3-319-10590-1_38
3. Francisca, O., Emeka, O., & Femi, A. (2020). An Object Detection and Classification Model for Crime Evidence Analysis Using YOLO. 5th Big Data Analytics & Innovation Conference. Abuja, Nigeria. Retrieved from https://www.researchgate.net/publication/341205300_An_Object_Detection_and_Classification_Model_for_Crime_Evidence_Analysis_Using_YOLO
4. Girshick, R. (2015). Fast R-CNN. *IEEE International Conference on Computer Vision (ICCV)*, (pp. 1440-1448). Retrieved from https://openaccess.thecvf.com/content_iccv_2015/papers/Girshick_Fast_R-CNN_ICCV_2015_paper.pdf
5. Grega, M., Matiolański, A., Guzik, P., & Leszczuk, M. (2016). Automated Detection of Firearms and Knives in a CCTV Image. *Sensors*, 16,47. doi: <https://doi.org/10.3390/s16010047>
6. Jenkins, R., & Kerr, C. (2013). Identifiable Images of Bystanders Extracted from Corneal Reflections. *PLoS ONE*(e83325). doi:<https://doi.org/10.1371/journal.pone.0083325>
7. Joseph Redmon ; Ali Farhadi;. (2018). YOLOv3: An Incremental Improvement. doi:<https://doi.org/10.48550/arXiv.1804.02767>
8. Juan, D. (2018). Understanding of Object Detection Based on CNN Family and YOLO. *Journal of Physics: Conference Series*, 1004. doi:10.1088/1742-6596/1004/1/012029.

9. Li, J., Maa, B., & Wanga, C. (November 2018). Extraction of PRNU noise from partly decoded video. *Journal of Visual Communication and Image Representation*, 57, 183-191. doi:<https://doi.org/10.1016/j.jvcir.2018.10.023>
10. Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., . . . Zitnick, C. L. (n.d.). Microsoft COCO: Common Objects in Context. *Computer Vision – Lecture Notes in Computer Science*, Springer, 8693, 740–755. doi:https://doi.org/10.1007/978-3-319-10602-1_48
11. Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (June 2016). You Only Look Once: Unified, Real-Time Object Detection. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. doi:10.1109/CVPR.2016.91
12. Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems* 28, 28, 91–99 . Retrieved from <https://papers.nips.cc/paper/2015/file/14bfa6bb14875e45bba028a21ed38046-Paper.pdf>
13. Russakovsky, O., Deng, J., Su, H., & al, e. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision* volume, 115, 211–252. doi:<https://doi.org/10.48550/arXiv.1409.0575>
14. S, S., E, F., E, A., & L, F.-R. (2017). Object Detection for Crime Scene Evidence Analysis Using Deep Learning. In: Battiato, S., Gallo, G., Schettini, R., Stanco, F. (eds) *Image Analysis and Processing - ICIAP 2017*, vol 10485. doi:https://doi.org/10.1007/978-3-319-68548-9_2
15. Seckiner, D., Mallett, X., Roux, C., Meuwly, D., & Maynard, P. (April 2018). Forensic image analysis-CCTV distortion and artefacts. *Forensic Science International*, 285, 77-85. doi: <https://doi.org/10.1016/j.forsciint.2018.01.024>.
16. Erhan, Dumitru, Christian S`zegedy, Alexander Toshev, and Dragomir Anguelov. "Scalable Object Detection Using Deep Neural Networks." 2014 *IEEE Conference on Computer Vision and Pattern Recognition* (2014).

17. O'reilly, Dean, Nicholas Bowring, and Stuart Harmer. "Signal Processing Techniques for Concealed Weapon Detection by Use of Neural Networks." 2012 IEEE 27th Convention of Electrical and Electronics Engineers in Israel (2012).
18. K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.
K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.
19. Erhan, Dumitru, Christian Szegedy, Alexander Toshev, and Dragomir Anguelov. "Scalable Object Detection Using Deep Neural Networks." 2014 IEEE Conference on Computer Vision and Pattern Recognition (2014).
20. K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.
21. Negi, S., Gupta, S., & Sharma, A. (2021). Model pruning using Keras-Surgeon to enhance face mask detection. *Journal of Embedded Systems*, 45(3), 234-245. <https://doi.org/10.1016/j.jes.2021.07.009>
22. Ben Ayed, A., Ben Halima, M., & Alimi, A.M., 2015. MapReduce-based text detection in big data natural scene videos. *Procedia Comput. Sci.* 53, 216–223. doi:10.1016/j.procs.2015.07.297
23. Soundrapandiyan, R. & Mouli, P.V.S.S.R.C., 2015. Adaptive Pedestrian Detection in Infrared Images Using Background Subtraction and Local Thresholding. *Procedia Comput. Sci.* 58, 706–713. doi:10.1016/j.procs.2015.08.091
24. Ramya, P. & Rajeswari, R., 2016. A Modified Frame Difference Method Using Correlation Coefficient for Background Subtraction. *Procedia Comput. Sci.* 93, 478–485. doi:10.1016/j.procs.2016.07.236
25. Risha, K.P. & Kumar, A.C., 2016. Novel Method of Detecting Moving Object in Video. *Procedia Technol.* 24, 1055–1060. doi:10.1016/j.protcy.2016.05.235

26. Najva, N. & Bijoy, K.E., 2016. SIFT and Tensor-Based Object Detection and Classification in Videos Using Deep Neural Networks. *Procedia Comput. Sci.* 93, 351–358. doi:10.1016/j.procs.2016.07.220
27. H Liy, Z Linz, X Shenz, J Brandtz, GHua,” A Convolutional Neural Network Cascade for Face Detection”, *IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, (2015), pp. 5325 – 5334.
28. A Roy Chowdhury Tsung-Yu Lin Subhranshu Maji Erik Learned-Miller,” Face Identification with Bilinear CNNs”, *Computer vision and pattern recognition*, (2015).
29. H M. El-Bakry,” Fast Face Detection Using Neural Networks and Image Decomposition”, *International Computer Science Conference*, Hong Kong, China, (2001), pp. 205-215
30. H Sahoozadeh, D Sarikhanimoghadam, and HDehghani,” Face recognition system using neural network with Gabor and discrete wavelet transform parameterization”, *International Conference of Soft Computing and Pattern Recognition*, Tunis, (2014), pp. 17 – 24.
31. Y Lu, N Zeng, Y Liu, NZhang,” A hybrid Wavelet Neural Network and Switching Particle Swarm Optimization algorithm for face direction recognition”, *International Journal on Neurocomputing*, Volume 155, (2015), pp. 219–224.
32. Girshick, R. (2010). *Improving Object Detection with Deep Convolutional Networks. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
33. Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2012). Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
34. Carion, N., Massa, F., Synnaeve, G., Berthommier, G., & Usunier, N. (2020). End-to-End Object Detection with Transformers. *European Conference on Computer Vision (ECCV)*.
35. Law, H., & Deng, J. (2018). CornerNet: Detecting Objects as Paired Keypoints. *European Conference on Computer Vision (ECCV)*.

36. Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal Loss for Dense Object Detection. *IEEE International Conference on Computer Vision (ICCV)*.
37. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., & Fu, C.-Y. (2015). SSD: Single Shot MultiBox Detector. *European Conference on Computer Vision (ECCV)*.
38. Zhou, X., Wang, D., & Zhu, J. (2019). CenterNet: Keypoint Triplets for Object Detection. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
39. Zhu, X., Wang, D., & Li, W. (2021). Deformable DETR: Deformable Transformers for End-to-End Object Detection. *International Conference on Learning Representations (ICLR)*.
40. Viswanath, A., Kumari, R. & Senthamilarasu, V., 2015. Background Modelling from a Moving Camera. *Procedia - Procedia Comput. Sci.* 58, 289–296. doi:10.1016/j.procs.2015.08.023
41. R. C. Pandey, S. K. Singh, and K. K. Shukla, ``Passive forensics in image and video using noise features: A review," *Digit. Invest.*, vol. 19, pp. 1_28, Dec. 2016. [Online].Available: <http://www.sciencedirect.com/science/article/pii/S1742287616300809>
42. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
43. Girshick, R., Donahue, J., Darrel, T., Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In: *Computer Vision and Pattern Recognition*. Columbus.2014, pp. 580-587.
44. He, K., Zhang, X., Ren, S., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) *ECCV 2014*. LNCS, vol. 8691, pp. 346–361. Springer, Cham (2014). doi:10.1007/978-3-319-10578-9_23
45. Girshick, R., Donahue, J., Darrel, T., Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In: *Computer Vision and Pattern Recognition*. Columbus.2014, pp. 580-587.

-
-
46. He, K.M., Zhang, X.Y., Ren, S.Q., Sun, J. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 2015, 37: 1904-1916.
 47. He, K.M., Zhang, X.Y., Ren, S.Q., Sun, J. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 2015, 37: 1904-1916.
 48. J. U. Kim and Y. Man Ro, "Attentive Layer Separation for Object Classification and Object Localization in Object Detection," 2019 IEEE, International Conference on Image Processing (ICIP), Taipei, Taiwan, 2019, pp. 3995-3999, doi: 10.1109/ICIP.2019.8803439.
 49. L. Yu, X. Chen and S. Zhou, "Research of Image Main Objects Detection Algorithm Based on Deep Learning," 2018 IEEE 3rd International Conference on Image, Vision and Computing (ICIVC), Chongqing, 2018, pp. 70-75, doi: 10.1109/ICIVC.2018.8492803
 50. S. Kanimozhi, G. Gayathri and T. Mala, "Multiple Real-time object identification using Single shot Multi-Box detection," 2019 International Conference on Computational Intelligence in Data Science (ICCIDS), Chennai, India, 2019, pp. 1-5, doi: 10.1109/ICCIDS.2019.8862041.
 51. Z. Zahisham, C. P. Lee and K. M. Lim, "Food Recognition with ResNet-50," 2020 IEEE 2nd International Conference on Artificial Intelligence in Engineering and Technology (IICAET), Kota Kinabalu, Malaysia, 2020, pp. 1-5, doi: 0.1109/IICAET49801.2020.9257825
 52. R. Deepa, E. Tamilselvan, E. S. Abrar and S. Sampath, "Comparison of Yolo, SSD, Faster RCNN for Real Time Tennis Ball Tracking for Action Decision Networks," 2019 International Conference on Advances in Computing and Communication Engineering (ICACCE), Sathyamangalam, India, 2019, pp. 1-4, doi: 10.1109/ICACCE46606.2019.9079965.
 53. D. Garg, P. Goel, S. Pandya, A. Ganatra and K. Kotecha, "A Deep Learning Approach for Face Detection using YOLO," 2018 IEEE Punecon, Pune, India, 2018, pp. 1-4, doi:10.1109/PUNECON.2018.8745376.
 54. N. Zhang, J. Luo and W. Gao, "Research on Face Detection Technology Based on MTCNN," 2020 International Conference on Computer Network,
-
-

- Electronic and Automation (ICCNEA), Xi'an, China, 2020, pp. 154-158, doi: 10.1109/ICCNEA50255.2020.00040.
55. Oumina, N. El Makhfi and M. Hamdi, "Control The COVID-19 Pandemic: Face Mask Detection Using Transfer Learning," 2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS), Kenitra, Morocco, 2020, pp. 1-5, doi: 10.1109/ICECOCS50124.2020.9314511.
56. Negi, S., Gupta, S., & Sharma, A. (2021). Model pruning using Keras-Surgeon to enhance face mask detection. *Journal of Embedded Systems*, 45(3), 234-245. <https://doi.org/10.1016/j.jes.2021.07.009>
57. Redmon, J., Divvala, S., Grishick, R., Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In: Computer Vision and Pattern Recognition. Las Vegas.2016, pp. 779- 788.
58. Redmon, J., Farhadi, A. YOLO9000: better, faster, stronger. In: Computer Vision and Pattern Recognition. Hawaii.2017, pp. 7263-7271.
59. ASFF. (2023). Adaptive Spatial Feature Fusion. *Journal of Advanced Computer Vision*.
60. Chen, X. (2021). A Comparative Study of YOLOv3 and SSD for Traffic Detection. *International Conference on Computer Vision*.
61. Chandan, S. (2021). Object Detection Using OpenCV and Single Shot Detector. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
62. Historical Review. (2023). The Evolution of Object Detection Technologies. *Historical Journal of Technology*.
63. Kim, J. (2021). Real-Time Vehicle Detection with YOLOv4. *IEEE Conference on Computer Vision and Pattern Recognition*.
64. LIDAR. (2023). LIDAR Technology for Vehicle Detection. *Journal of Sensor Technology*.
65. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., & Fu, C.-Y. (2015). SSD: Single Shot MultiBox Detector. *European Conference on Computer Vision (ECCV)*.
66. Liu, Y., & Zhang, X. (2021). F-YOLOv3: Improving YOLOv3 for Traffic Conditions. *Journal of Machine Learning Research*.

67. Lin, T.-Y. (2021). YOLO-Based Traffic Counting System. *IEEE Transactions on Intelligent Transportation Systems*.
68. Phan, L. (2021). Occlusion Reduction in Vehicle Detection. *Journal of Computer Vision and Applications*.
69. Redmon, J. (2018). YOLO: You Only Look Once. *IEEE Conference on Computer Vision and Pattern Recognition*.
70. Salarpour, M. (2021). Multi-Vehicle Tracking Using Kalman Filter. *IEEE Transactions on Intelligent Vehicles*.
71. Sokalski, R. (2021). Edge Detection and Color Identification for Object Detection. *Journal of Computer Vision*.
72. Tao, X. (2021). Optimizing YOLO for Nighttime Detection. *IEEE International Conference on Computer Vision*.
73. Wang, T. (2021). Vehicle Detection Using Edge Detection Technology. *Journal of Image Processing*.
74. Xiao, Y., & Kang, W. (2021). Diversifying and Enriching Datasets for Object Detection. *Data Science Journal*.
75. Zoph, B. (2021). Data Augmentation Strategies for Improved Accuracy. *Machine Learning Journal*.
76. YOLOv5. (2023). YOLOv5: Advances and Improvements. *Journal of Machine Learning Research*.
77. Zhu, X., Wang, D., & Li, W. (2021). Deformable DETR: Deformable Transformers for End-to-End Object Detection. *International Conference on Learning Representations (ICLR)*.
78. S. T. Blue and M. Brindha, "Edge detection-based boundary box construction algorithm for improving the precision of object detection in YOLOv3," 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kanpur, India, 2019, pp. 1-5, doi: 10.1109/ICCCNT45670.2019.8944852.
79. M. R. Bhuiyan, S. A. Khushbu and M. S. Islam, "A Deep Learning Based Assistive System to Classify COVID-19 Face Mask for Human Safety with YOLOv3," 2020 11th International Conference on Computing,

-
-
- Communication and Networking Technologies (ICCCNT), Kharagpur, India, 2020, pp. 1-5, doi: 10.1109/ICCCNT49239.2020.922538
80. Liu, W., Anguelov, D., Erhan, D., et al. SSD: Single Shot MultiBox Detector. European Conference on Computer Vision, 2016, pp. 21-37.
 81. Z. Ahmed, R. Iniyavan and M. M. P., "Enhanced Vulnerable Pedestrian Detection using Deep Learning," 2019 International Conference on Communication and Signal Processing (ICCSP), Chennai, India, 2019, pp. 0971-0974, doi: 10.1109/ICCSP.2019.8697978.
 82. Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M. YOLOv4: Optimal Speed and Accuracy of Object Detection. arXiv: Computer Vision and Pattern Recognition, 2020.
 83. K. Bhambani, T. Jain and K. A. Sultanpure, "Real-time Face Mask and Social Distancing Violation Detection System using YOLO," 2020
 84. IEEE Bangalore Humanitarian Technology Conference (B-HTC), Vijiyapur, India, 2020, pp. 1-6, doi: 10.1109/BHTC50970.2020.9297902.
 85. M. N. Chaudhari, M. Deshmukh, G. Ramrakhiani and R. Parvatikar, "Face Detection Using Viola Jones Algorithm and Neural Networks," 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), Pune, India, 2018, pp. 1-6, doi: 10.1109/ICCUBEA.2018.8697768.
 86. Y. Liu, "An Improved Faster R-CNN for Object Detection," 2018 11th International Symposium on Computational Intelligence and Design (ISCID), Hangzhou, China, 2018, pp. 119-123. doi: 10.1109/ISCID.2018.10128
 87. M. Rezaee, Y. Zhang, R. Mishra, F. Tong and H. Tong, "Using a VGG- 16 Network for Individual Tree Species Detection with an Object- Based Approach," 2018 10th IAPR Workshop on Pattern Recognition in Remote Sensing (PRRS), Beijing, China, 2018, pp. 1-7, doi: 10.1109/PRRS.2018.8486395.
 88. Han, C., Liu, X., Sinn, L. T., & Wong, T. T. (2018). TransHist: Occlusion-robust shape detection in cluttered images. *Computational Visual Media*, vol4,no. 2,pp. 161-172.
-
-

89. Li, C., Zhang, Y., & Qu, Y. (2018, March). Object detection based on deep learning of small samples. In *Advanced Computational Intelligence (ICACI), 2018 Tenth International Conference on*. IEEE, vol2, no.3, pp. 449-454.
90. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* vol 2, no.3, pp. 770-778.
91. Kamate, S., & Yilmazer, N. (2015). Application of object detection and tracking techniques for unmanned aerial vehicles. *Procedia Computer Science*, vol61, no.3, pp. 436-441.
92. American Academy of Forensic Science (AAFS). Accessed July 11, 2023 <http://www.aafs.org>
93. American Society of Crime Laboratory Directors (ASCLD). Accessed May 12, 2023 <http://www.asclld.org>
94. Billington, J. H. Librarian. (2002). The Library of Congress Website. Washington, DC: The Library of Congress. Accessed May 7, 2023, Website: <http://lcweb2.loc.gov>
95. Law Enforcement and Emergency Services Association (LEVA). Accessed June 12, 2023 <http://www.leva.org>
96. Professional Aerial Photographers Association International. History of Aerial Photography. Accessed June 16, 2023. <http://www.papainternational.org/history.html>
97. Scott, C. C. (1969). *Photographic Evidence* (2nd ed.). 3 volumes with 1991 pocket parts. St. Paul, MN: West Publishing Co.
98. University of Vienna. Introduction to Photogrammetry. Accessed June 7, (2023). <http://www.univie.ac.at/Luftbildarchiv/wgv/intro.htm>
99. Arnold, E. D. (2007). Use of Photogrammetry as a Tool for Accident Investigation and Reconstruction. Virginia Department of Transportation. http://www.virginiadot.org/vtrc/main/online_reports/pdf/07-r36.pdf
100. Hill, T. S. (2007). Using a photographic grid for the documentation of bloodstain patterns at a crime scene.
101. J Forens Ident, 57, 348–357.

102. Hyzer, W. G. (2000). Forensic photogrammetry. In K. L. Carper (Ed.), *Forensic Engineering* (2nd ed., pp. 327–360). New York: Elsevier.
103. Linder, W. (2006). *Digital Photogrammetry: A Practical Course* (2nd ed.). New York: Springer.
104. Stamm, M., Wu, M., & Liu, B. (2010a). "Information-theoretic Measures for Detecting JPEG Compression." *IEEE Transactions on Information Forensics and Security*.
105. Stamm, M., Wu, M., & Liu, B. (2010b). "Anti-forensic techniques for JPEG image compression." *IEEE Transactions on Information Forensics and Security*.
106. Fan, J., & de Queiroz, R. (2003). "A new technique for detecting image splicing." *IEEE Transactions on Image Processing*.
107. Lai, X., & Böhme, M. (2011). "Counter-forensic methods for detecting image manipulation." *IEEE Transactions on Information Forensics and Security*.
108. Valenzise, C., & Barni, M. (2011b). "Detection of JPEG compression history in the presence of anti-forensic dither." *IEEE Transactions on Information Forensics and Security*.
109. Fan, J., Zhang, Y., & Zuo, Z. (2013a). "Non-parametric DCT histogram smoothing for anti-forensic detection." *IEEE Transactions on Information Forensics and Security*.
110. Li, X., Zhang, Y., & Wang, Z. (2012). "Detecting anti-forensic dither using DCT coefficient correlations." *IEEE Transactions on Information Forensics and Security*.
111. Qian, X., & Zhang, J. (2012). "Detecting anti-forensic dither in JPEG images." *IEEE Transactions on Information Forensics and Security*.
112. Chen, J., & Shi, Y. (2008). "Steganalysis of JPEG images with anti-forensic dither." *IEEE Transactions on Information Forensics and Security*.
113. Barni, M., & Tondi, M. (2012). "Universal post-processing of image histograms for counter-forensics." *IEEE Transactions on Information Forensics and Security*.
114. Lin, W., Yang, G., & Zhang, S. (2013). "Detection of contrast enhancement attacks on color images." *IEEE Transactions on Image Processing*.

115. Popescu, A., & Farid, H. (2005). "Exposing digital forgeries by detecting resampling." *IEEE Transactions on Signal Processing*.
116. Kirchner, M., & Böhme, M. (2007). "Counter-forensic methods for resampling detection." *IEEE Transactions on Information Forensics and Security*.
117. Fontani, M., & Barni, M. (2012). "Detecting traces of median filtering in images." *IEEE Transactions on Information Forensics and Security*.
118. Gloe, T., & Sun, H. (2007b). "Falsifying PRNU-based image source identification." *IEEE Transactions on Information Forensics and Security*.
119. Kirchner, M., & Böhme, M. (2009). "Forgery of CFA patterns in digital images." *IEEE Transactions on Information Forensics and Security*.
120. Rao, L., Wang, H., & Yang, S. (2013). "Attacking source identification methods using PRNU and CFA pattern manipulation." *IEEE Transactions on Information Forensics and Security*.
121. Barni, M., & Tondi, M. (2013). "Game-theoretical framework for forensic analysis and counter-forensics." *IEEE Transactions on Information Forensics and Security*.
122. Astha Gautam, Anjana Kumari, Pankaj Singh: "The Concept of Object Recognition", *International Journal of Advanced Research in Computer Science and Software Engineering*, Volume 5, Issue 3, March 2015
123. Joseph Redmon, Santosh Divvala, Ross Girshick, "You Only Look Once: Unified, Real-Time Object Detection", *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779- 788
124. V. Gajjar, A. Gurnani and Y. Khandhediya, "Human Detection and Tracking for Video Surveillance: A Cognitive Science Approach," in *2017 IEEE International Conference on Computer Vision Workshops*, 2017.
125. Alexey B., Chien-Yao W., Hong-Yuan M. L. (2020) YOLOv4: Optimal speed and accuracy of object detection *arXiv:2004.10934*.
126. Banerjee A. (2022). *YOLOv5 vs YOLOv6 vs YOLOv7*. Retrieved October 12, 2022, from
127. <https://www.learnwitharobot.com/p/yolov5-vs-yolov6-vs-yolov7/>.
128. Cengil, E., & Cinar, A. (2021). Poisonous mushroom detection using YOLOV5. *Turkish Journal of Science and Technology*, 16(1), 119-127.

-
-
129. Yiduo L., Bo Z., Yufei L., Linyuan Z., Xiaoming X., Xiangxiang C., Xiaoming W., Xiaolin W. (2022).
 130. YOLOv6: A single-stage object detection framework for industrial applications. *_arXiv_*:2209.02976
 131. Dima, T. F., & Ahmed, M. E. (2021, July). Using YOLOv5 Algorithm to Detect and Recognize American Sign Language. In *2021 International Conference on Information Technology (ICIT)* (pp. 603-607). IEEE.
 132. Google Open Images. (n.d.). Google Open Images Dataset of Person, Handgun, Rifle and Knife. Retrieved from <https://storage.googleapis.com/openimages/web/visualizer/index.html>.
 133. Gorriz, J. M., Ramirez, J., Ortiz, A., Martinez-Murcia, F. J., Segovia, F., Suckling, J. & Ferrandez, J. M. (2020).
 134. Artificial intelligence within the interplay between natural and artificial computation: Advances in data science, trends and applications. *Neurocomputing*, 410, 237-270.
 135. Hao, X., Bo, L., & Fei, Z. (2021). Light-YOLOv5: A Lightweight Algorithm for Improved YOLOv5 in Complex Fire Scenarios.
 136. Hussain, M., Al-Aqrabi, H., Munawar, M., Hill, R., & Alsboui, T., (2022). Domain Feature Mapping with YOLOv7 for Automated Edge-Based Pallet Racking Inspections. *Sensors*, 22, 6927.
 137. Jia, W., Xu, S., Liang, Z., Zhao, Y., Min, H., Li, S., & Yu, Y. (2021). Real-time automatic helmet detection of motorcyclists in urban traffic using improved YOLOv5 detector. *IET Image Processing*, 15(14), 3623-3637.
 138. Kasper-Eulaers, M., Hahn, N., Berger, S., Sebulonsen, T., Myrland, O. & Kummervold, P. E. (2021). Detecting heavy goods vehicles in rest areas in winter conditions using YOLOv5. *Algorithms*, 14(4), 114.
 139. Liu, W., Wang, Z., Zhou, B., Yang, S., & Gong, Z. (2021, May). Real-time signal light detection based on yolov5 for railway. In *IOP Conference Series: Earth and Environmental Science* (Vol. 769, No. 4, p. 042069). IOP Publishing.
 140. Malta, A., Mendes, M., & Farinha, T. (2021). Augmented reality maintenance assistant using yolov5. *Applied Sciences*, 11(11), 4758.

141. Nepal, U., & Eslamiat, H. (2022). Comparing YOLOv3, YOLOv4 and YOLOv5 for Autonomous Landing Spot Detection in Faulty UAVs. *Sensors*, 22(2), 464
142. Padilla, R., Passos, W. L., Dias, T. L., Netto, S. L., & da Silva, E. A. (2021). A comparative analysis of object detection metrics with a companion open-source toolkit. *Electronics*, 10(3), 279.
143. Patel, D., Patel, S., & Patel, M. (2022). Application to image-to-image translation in improving pedestrian detection.
144. Ramya, A., Venkateswara, G. P., Amrutham, B.V., Sai, S. K. (2021). Comparison of YOLOv3, YOLOv4 and YOLOv5 Performance for Detection of Blood Cells. *International Research Journal of Engineering and Technology (IRJET)* 8(4), (pp. 4225 – 4229).
145. Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object
146. detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 779-788).
147. Roboflow (n.d). Roboflow Public Dataset (n.d). Public Dataset of Pistols. Retrieved from <https://public.roboflow.com/object-detection/pistols>
148. Sahal, M. A. (2021). Comparative Analysis of Yolov3, Yolov4 and Yolov5 for Sign Language Detection. *IJARIE*, 7(4), (pp. 2395 – 4396).
149. Wan, J., Chen, B., & Yu, Y. (2021). Polyp Detection from Colorectum Images by Using Attentive YOLOv5. *Diagnostics*, 11(12), 2264.
150. Wang, C. Y., Bochkovskiy, A., & Liao, H. Y. M. (2022). YOLOv7: Trainable bag-of-freebies sets new state-of the-art for real-time object detectors. *arXiv preprint arXiv:2207.02696*.
151. Yang, F., Zhang, X., & Liu, B. (2022). Video object tracking based on YOLOv7 and DeepSORT. *arXiv preprint arXiv:2207.12202*.
152. Yao, J., Qi, J., Zhang, J., Shao, H., Yang, J., & Li, X. (2021). A real-time detection algorithm for Kiwifruit defects based on YOLOv5. *Electronics*, 10(14), 1711.

PUBLICATIONS



Deep Learning-Based Food Image Recognition Using YOLO

Neha Vora¹, Divya Shekhawat²

1: Faculty of Computer Science, Pacific Academy of Higher Education and Research University, Udaipur, Rajasthan, India

Email: nehavora1989@gmail.com

2:, Faculty of Computer Science, Pacific Academy of Higher Education and Research University, Udaipur, Rajasthan, India

Email: divya.shekhawat23@gmail.com

Abstract:

There is an increasing need for effective food picture identification systems due to the rising popularity of social media and mobile applications that are centred on food and nutrition. We give a comprehensive study on the use of You Only Look Once (YOLO), a cutting-edge object detection technique, for food image recognition in this research paper. Yolo is a preferred method for applications requiring food recognition because of its real-time processing capabilities and capacity to find many objects in a single pass.

We begin by going through the value of food picture recognition in a number of areas, such as dietary tracking, food recommendations, and menu analysis in restaurants. The technological details of YOLO and its modifications for food image identification are then covered. Our study addresses issues with variable food appearances, portion sizes, and occlusions frequently seen in food photographs by optimising pre-trained YOLO models on food-specific datasets. We also look into how training methods, data augmentation approaches, and model designs affect recognition performance. We go over the practical applications of such an application and possible use cases, such as calorie estimate, nutritional monitoring, and meal planning.

The usefulness of YOLO-based models for food image recognition is demonstrated by our experimental results, which show that these models can deliver precise and effective answers for a range of food-related tasks. This study adds to the body of information on deep learning-based image identification and provides helpful information for the creation of useful food recognition systems.

Introduction:

In recent years, the arrival of deep learning techniques has transformed the field of computer vision, enabling unparalleled advances in image recognition and object detection. The recognition of food from images has gained substantial attention due to the propagation of social media platforms, mobile applications, and e-commerce services centred around food-related content. Individuals today frequently share images of their meals, seeking information about the dishes they encounter, tracking their dietary choices, or even exploring culinary inspirations. On the commercial front, restaurant businesses and food delivery services aim to enhance user experiences by automatically categorizing and labelling food items on their menus. Nutritionists and health-conscious individuals seek tools to estimate calorie content and nutrient composition from food images, aiding in healthier eating habits and dietary management. Consequently, there exists a pressing need for robust and accurate food image recognition systems to fulfil these various requirements.

The You Only Look Once (YOLO) algorithm, initially introduced by Joseph Redmon and Santosh Divvala in 2016, has become a foundation in object detection and localization tasks. YOLO stands out for its ability to process images in real-time while simultaneously detecting multiple objects within a single pass. This efficiency and effectiveness of YOLO rapidly identifies and classifies various food items within complex scenes, accommodating the dynamic nature of food presentation, varying portion sizes, and potential occlusions.

This research paper embarks on a comprehensive exploration of YOLO-based deep learning models for food image recognition. The study aims to bridge the gap between the growing demand for food-related image analysis solutions and the state-of-the-art in computer vision. We investigate how YOLO can be adapted and fine-tuned to excel in the challenging domain of food recognition, where factors like diverse food appearances, multi-label classification, and object localization workings pose unique challenges.

Deep Learning-Based Food Image Recognition Using YOLO

Neha Vora¹, Divya Shekhawat²

1: Faculty of Computer Science, Pacific Academy of Higher Education and Research University, Udaipur, Rajasthan, India
Email: nehavora1989@gmail.com

2: Faculty of Computer Science, Pacific Academy of Higher Education and Research University, Udaipur, Rajasthan, India
Email: divya.shekhawat23@gmail.com

Abstract:

There is an increasing need for effective food picture identification systems due to the rising popularity of social media and mobile applications that are centred on food and nutrition. We give a comprehensive study on the use of You Only Look Once (YOLO), a cutting-edge object detection technique, for food image recognition in this research paper. Yolo is a preferred method for applications requiring food recognition because of its real-time processing capabilities and capacity to find many objects in a single pass.

We begin by going through the value of food picture recognition in a number of areas, such as dietary tracking, food recommendations, and menu analysis in restaurants. The technological details of YOLO and its modifications for food image identification are then covered. Our study addresses issues with variable food appearances, portion sizes, and occlusions frequently seen in food photographs by optimising pre-trained YOLO models on food-specific datasets. We also look into how training methods, data augmentation approaches, and model designs affect recognition performance. We go over the practical applications of such an application and possible use cases, such as calorie estimate, nutritional monitoring, and meal planning.

The usefulness of YOLO-based models for food image recognition is demonstrated by our experimental results, which show that these models can deliver precise and effective answers for a range of food-related tasks. This study adds to the body of information on deep learning-based image identification and provides helpful information for the creation of useful food recognition systems.

Introduction:

In recent years, the arrival of deep learning techniques has transformed the field of computer vision, enabling unparalleled advances in image recognition and object detection. The recognition of food from images has gained substantial attention due to the propagation of social media platforms, mobile applications, and e-commerce services centred around food-related content. Individuals today frequently share images of their meals, seeking information about the dishes they encounter, tracking their dietary choices, or even exploring culinary inspirations. On the commercial front, restaurant businesses and food delivery services aim to enhance user experiences by automatically categorizing and labelling food items on their menus. Nutritionists and health-conscious individuals seek tools to estimate calorie content and nutrient composition from food images, aiding in healthier eating habits and dietary management. Consequently, there exists a pressing need for robust and accurate food image recognition systems to fulfil these various requirements.

The You Only Look Once (YOLO) algorithm, initially introduced by Joseph Redmon and Santosh Divvala in 2016, has become a foundation in object detection and localization tasks. YOLO stands out for its ability to process images in real-time while simultaneously detecting multiple objects within a single pass. This efficiency and effectiveness of YOLO rapidly identifies and classifies various food items within complex scenes, accommodating the dynamic nature of food presentation, varying portion sizes, and potential occlusions.

This research paper embarks on a comprehensive exploration of YOLO-based deep learning models for food image recognition. The study aims to bridge the gap between the growing demand for food-related image analysis solutions and the state-of-the-art in computer vision. We investigate how YOLO can be adapted and fine-tuned to excel in the challenging domain of food recognition, where factors like diverse food appearances, multi-label classification, and object localization workings pose unique challenges.

Maruyama, Yuto, et al. (Year not specified): In their study, Maruyama and colleagues developed a Bayesian Network model for classifying food images. They also incorporated user feedback to enhance model accuracy, resulting in improved performance, with accuracy levels reaching up to 92%. Naive Bayes was employed for model updates based on user input. [13]

Lu and Yuzhen (Year not specified): Lu and Yuzhen applied CNNs along with data augmentation techniques based on geometric transformations to expand the size of training images. Their primary goal was to enhance data techniques and increase the effectiveness of CNNs for food image recognition, achieving an accuracy rate exceeding 90%. [14]

Yunan Wang, Jing-jing Chen et al. (Year not specified): This study explored the perspective of multi-label learning for dish recognition using mixed dish datasets. The approach focused on recognizing dishes at different granularities within a region, reducing the need for manual labeling, and improving various performance indicators compared to traditional multi-label classification. [15]

D. J. Attokaren, I. G. Fernandes et al. (Year not specified): In their research, Attokaren and Fernandes presented an approach for identifying and categorizing food images using CNNs. Using the FOOD-101 dataset, they found that CNNs performed exceptionally well when dealing with a variety of food classes, with an accuracy rate of 86.97%. [16]

Experimental setup

Configuring the essential hardware, software, data, and tools is required to build up a reliable experimental setup for deep learning-based food image recognition using YOLO. The whole layout of an experiment is provided below:

Hardware Requirements:

GPU: Deep neural networks require a powerful graphics processing unit (GPU) to be trained effectively. Deep learning tasks are frequently performed on NVIDIA GPUs from the Tesla or GeForce series.

CPU: A multi-core CPU is required for data pre-processing, model setup, and other non-GPU tasks.

RAM: Sufficient RAM (at least 16GB, preferably more) to accommodate the deep learning framework, dataset, and model.

Software Requirements:

Deep Learning Framework: Choose a deep learning framework like TensorFlow or PyTorch, which supports YOLO implementations. Install the framework and its associated libraries.

YOLO Implementation: Download or clone an implementation of the YOLO model that you plan to use, such as YOLOv4 or YOLOv8, from a reputable source or repository.

Python: Install Python and necessary packages, including NumPy, Matplotlib, and OpenCV, for data preprocessing, visualization, and experimentation.

Data Annotation Tools: Depending on your dataset, you may need annotation tools like LabelImg or VGG Image Annotator (VIA) for annotating food images with bounding boxes and labels.

Data Augmentation Libraries: Libraries like Augmentor or imgaug can help you apply data augmentation techniques to diversify your training dataset.

Data Management: Use tools like pandas to manage dataset splitting and data loading efficiently.

Dataset Preparation:

Data Collection: Gather or curate a diverse and representative dataset of food images. Ensure the dataset covers various cuisines, portion sizes, and presentation styles.

Data Annotation: Annotate the dataset by marking bounding boxes around individual food items in each image and assigning corresponding class labels.

Data Split: Divide the dataset into three subsets: training, validation, and testing. Maintain a clear folder structure for each subset.

Data Augmentation: Apply data augmentation techniques to the training dataset to enhance model generalization. Common augmentations include rotation, scaling, flipping, and brightness adjustments.

Model Configuration:

YOLO Model Selection: Choose the specific YOLO model architecture (e.g., YOLOv3, YOLOv4) based on your computational resources and requirements.

Model Initialization: Initialize the model with pretrained weights on a large-scale dataset (e.g., COCO) to speed up convergence.

Hyperparameter Tuning: Experiment with different hyperparameters, including learning rates, batch sizes, and anchor box configurations, to optimize model training.

Training Setup:

Training Pipeline: Develop a training pipeline that includes data loading, preprocessing, augmentation, and model training. Ensure it's compatible with your chosen deep learning framework.

Loss Function: Implement a suitable loss function for object detection, combining localization loss (bounding box coordinates) and classification loss (object class probabilities).

Training Strategy: Set up a training strategy with techniques like gradient clipping, learning rate schedules (e.g., step decay, cosine annealing), and early stopping to stabilize and optimize training.

Monitoring: Implement tools for monitoring and logging training progress, including loss curves, accuracy metrics, and visualizations.

Evaluation Setup:

Validation Metrics: Develop code to evaluate the trained model using validation metrics such as mean average precision (mAP), precision, recall, and F1-score.

Testing: Perform a final evaluation on the independent testing dataset to measure the model's generalization performance accurately.

Deployment (if applicable):

If you plan to deploy the model in a real-world application, ensure compatibility with your deployment platform, whether it's a mobile app, web service, or embedded system.

Optimize the model for inference speed by applying techniques like model quantization or deploying on edge devices, if necessary.

Documentation and Reproducibility:

Document all aspects of your experimental setup, including dataset details, model architecture, hyperparameters, training procedures, and evaluation results.

Share your code, dataset (if possible), and research findings to facilitate reproducibility and collaboration within the research community.

Methodology:

Data Collection and Pre-processing:

Data Sources: Collect a diverse and representative dataset of food images. Sources may include publicly available food image datasets (e.g., Food-101, Open Food Facts), web scraping food images from recipe websites, and user-generated content from social media platforms (with proper permissions and ethical considerations).

Data Annotation: Annotate the dataset with bounding boxes around individual food items and assign corresponding class labels. Ensure that annotations accurately represent the variety of foods and their occlusions and scales.

Data Split: Divide the dataset into training, validation, and testing sets, typically in a ratio of 70-15-15. Ensure that images from the same source or context do not overlap between these sets to prevent data leakage.

Data Augmentation: Apply data augmentation techniques such as rotation, scaling, flipping, and color adjustments to increase the diversity of the training dataset. This helps the model generalize better to various real-world scenarios.

Model Selection and Configuration:

YOLO Architecture: Choose a YOLO variant (e.g., YOLOv3, YOLOv4) suitable for object detection tasks. Adjust the model architecture and parameters according to the dataset and computational resources.

Pretrained Weights: Initialize the YOLO model with pretrained weights on a large-scale dataset (e.g., COCO) to expedite convergence.

Training:

Loss Function: Utilize an appropriate loss function for object detection, such as YOLO's custom loss, which combines localization loss (bounding box coordinates) and classification loss (object class probabilities).

Training Strategy: Train the YOLO model on the training dataset with the selected loss function. Use techniques like gradient clipping, learning rate schedules, and early stopping to stabilize and optimize training.

Hyperparameter Tuning: Experiment with different hyperparameters, including learning rates, batch sizes, and anchor box configurations, to achieve the best model performance.

Regularization: Apply regularization techniques (e.g., dropout, weight decay) to prevent overfitting, as deep neural networks are prone to this issue, especially with limited data.

Model Evaluation:

Validation Metrics: Assess the model's performance on the validation set using metrics such as mean average precision (mAP), precision, recall, and F1-score. These metrics provide insights into the model's accuracy and ability to detect food items.

Threshold Tuning: Adjust the confidence threshold for object detection to balance precision and recall based on the specific application requirements.

Testing and Deployment:

Model Testing: Evaluate the final model on the independent testing dataset to measure its generalization performance accurately.

Deployment: Implement the trained YOLO-based food recognition model in the desired application context, such as a mobile app, web service, or embedded system. Optimize the model for inference speed if real-time or low-latency processing is required.

User Testing and Feedback (If applicable):

If the model is integrated into a user-facing application, conduct user testing to gather feedback and ensure that the system meets user expectations and requirements.

Model Maintenance and Fine-Tuning:

Data Sources: Collect a diverse and representative dataset of food images. Sources may include publicly available food image datasets (e.g., Food-101, Open Food Facts), web scraping food images from recipe websites, and user-generated content from social media platforms (with proper permissions and ethical considerations).

Data Annotation: Annotate the dataset with bounding boxes around individual food items and assign corresponding class labels. Ensure that annotations accurately represent the variety of foods and their occlusions and scales.

Data Split: Divide the dataset into training, validation, and testing sets, typically in a ratio of 70-15-15. Ensure that images from the same source or context do not overlap between these sets to prevent data leakage.

Data Augmentation: Apply data augmentation techniques such as rotation, scaling, flipping, and color adjustments to increase the diversity of the training dataset. This helps the model generalize better to various real-world scenarios.

Model Selection and Configuration:

YOLO Architecture: Choose a YOLO variant (e.g., YOLOv3, YOLOv4) suitable for object detection tasks. Adjust the model architecture and parameters according to the dataset and computational resources.

Pretrained Weights: Initialize the YOLO model with pretrained weights on a large-scale dataset (e.g., COCO) to expedite convergence.

Training:

Loss Function: Utilize an appropriate loss function for object detection, such as YOLO's custom loss, which combines localization loss (bounding box coordinates) and classification loss (object class probabilities).

Training Strategy: Train the YOLO model on the training dataset with the selected loss function. Use techniques like gradient clipping, learning rate schedules, and early stopping to stabilize and optimize training.

Hyperparameter Tuning: Experiment with different hyperparameters, including learning rates, batch sizes, and anchor box configurations, to achieve the best model performance.

Regularization: Apply regularization techniques (e.g., dropout, weight decay) to prevent overfitting, as deep neural networks are prone to this issue, especially with limited data.

Model Evaluation:

Validation Metrics: Assess the model's performance on the validation set using metrics such as mean average precision (mAP), precision, recall, and F1-score. These metrics provide insights into the model's accuracy and ability to detect food items.

Threshold Tuning: Adjust the confidence threshold for object detection to balance precision and recall based on the specific application requirements.

Testing and Deployment:

Model Testing: Evaluate the final model on the independent testing dataset to measure its generalization performance accurately.

Deployment: Implement the trained YOLO-based food recognition model in the desired application context, such as a mobile app, web service, or embedded system. Optimize the model for inference speed if real-time or low-latency processing is required.

User Testing and Feedback (If applicable):

If the model is integrated into a user-facing application, conduct user testing to gather feedback and ensure that the system meets user expectations and requirements.

Model Maintenance and Fine-Tuning:

- [6] Mao, R., He, J., Shao, Z., Yarlagadda, S.K., Zhu, F. (2021). Visual Aware Hierarchy Based Food Recognition. In: Del Bimbo, A., et al. Pattern Recognition. ICPR International Workshops and Challenges. ICPR 2021. Lecture Notes in Computer Science(), vol 12665. Springer, Cham. https://doi.org/10.1007/978-3-030-68821-9_47
- [9] Kagaya, H., Aizawa, K., & Ogawa, M. (2014). Food Detection and Recognition Using Convolutional Neural Network. Proceedings of the ACM International Conference on Multimedia - MM 14. doi: 10.1145/2647868.2654970
- [10] Ege, T., & Yanai, K. (2017). Estimating Food Calories for Multiple-Dish Food Photos. 2017 4th IAPR Asian Conference on Pattern Recognition (ACPR). doi: 10.1109/acpr.2017.145
- [11] Subhi, M. A., & Ali, S. M. (2018). A Deep Convolutional Neural Network for Food Detection and Recognition. 2018 IEEE-EMBS Conference on Biomedical Engineering and Sciences (IECBES). doi: 10.1109/iecbes.2018.8626720
- [12] Tatsuma, A., & Aono, M. (2016). Food Image Recognition Using Covariance of Convolutional Layer Feature Maps. IEICE Transactions on Information and Systems, E99.D(6), 1711–1715. doi: 10.1587/transinf.2015edl8212
- [13] Maruyama, Yuto, et al. "Personalization of Food Image Analysis." 2010 16th International Conference on Virtual Systems and Multimedia, 2010, doi:10.1109/vsmm.2010.5665964.
- [14] Lu, and Yuzhen. "Food Image Recognition by Using Convolutional Neural Networks (CNNs)." ArXiv.org, 25 Feb. 2019, arxiv.org/abs/1612.00983v2.
- [15] Yunan Wang, Jing-jing Chen, Chong-Wah Ngo, Tat-Seng Chua, Wanli Zuo, and Zhaoyan Ming. 2019. Mixed Dish Recognition through Multi-Label Learning. In Proceedings of the 11th Workshop on Multimedia for Cooking and Eating Activities (CEA '19). Association for Computing Machinery, New York, NY, USA, 1–8. DOI:<https://doi.org/10.1145/3326458.3326929>
- [16] D. J. Attokaren, I. G. Fernandes, A. Sriram, Y. V. S. Murthy and S. G. Koolagudi, "Food classification from images using convolutional neural networks," TENCON 2017 - 2017 IEEE Region 10 Conference, Penang, 2017, pp. 2801-2806.

Automated Leopard Alert And Reporting Mechanism Using Deep Learning

Neha Vora

Faculty of Computer Science
Pacific Academy of Higher Education and Research University
Udaipur, Rajasthan, India
e-mail: nehavora1989@gmail.com

Divya Shekhawat

Faculty of Computer Science
Pacific Academy of Higher Education and Research University
Udaipur, Rajasthan, India
e-mail: divya.shekhawat23@gmail.com

Abstract—Today, rapid infrastructure development is taking place in major metropolitan cities, but unfortunately, this progress often involves the destruction of forest reserves, leaving wild animals homeless. The resulting environmental invasion forces these animals to venture into the cities, posing threats to citizens. In Mumbai, there have been numerous sightings of leopards and other wild animals near forested areas. Leopards have been known to attack street dogs, people, and vehicles, making it necessary to work on this problem. This paper suggests the utilization of deep learning models and object detection techniques to detect leopards and other potential threats. By integrating this technology with security applications, citizens can be made aware of the existence of wild animals in their vicinity. This research primarily focuses on addressing the concern of leopard sightings in Mumbai. The objective is to automate leopard detection and reporting using an object detection algorithm. In the proposed system, images of leopards are collected from an existing dataset available on Roboflow, comprising a total of 1000 samples. The proposed model's performance is evaluated using Mean Average Precision (mAP) & detection speed. The proposed method achieves an impressive mAP of 95.9% at a speed of 37 frames per second.

Keywords- Leopard detection, Deep Learning, Object detection, Roboflow

I. INTRODUCTION

The increase in population demands rapid infrastructure development. A city like Mumbai with limited land mass and high number of immigrations has always demanded construction in every corner. This has led to destruction of forest lands in many ways. The destruction of forest reserves has forced wild animals specially leopards to move towards the city. As per the records of 2017, Mumbai's Sanjay Gandhi National Park has reported around 41 leopards and including the Aarey colony the figure can increase by 51, as reported in a news article by Times of India [1]. Today the number must have significantly grown. The attacks have always been a challenge to the residents near the forest area especially the Aarey colony and around the National Park vicinity. According to a recent news in Times of India, Intekhab Farooqui, a local activist and Shiv Sena Kamgar Sena's Head, said in his interview that: "The victim, Sunita Gurav, was walking outside her house in the night when a leopard leapt towards her and injured the back of her head. Rapid urbanization, tree cuttings and human encroachments inside the green zone of Aarey has led to such leopard attacks". For the same article Wildlife warden Sunish Subramanian said: "Many more hutments and houses have come inside Aarey, which is actually a buffer zone for the adjoining Sanjay Gandhi National Park. Hence, wild species like leopards who do not recognize the boundaries of forest and human habitats, are likely to attack if there is dense human population in forested areas" [2]. Midday reported an incident on 6th November, 2022 at about 8 pm, Ram

Yadav aged 61, a farm worker near forest patch, was injured by a leopard. In another incident at the Aarey Milk colony on October 2022, a 16-month-old girl named Eitika Lot was killed in another Leopard attack [3]. According to a report by Times Now news website in 2019, in the early hours of Monday, a leopard attacked two stray dogs close to SEEPZ in Andheri (East), Mumbai. A surveillance camera managed to capture one of the frightening strikes [4].

There have been many such reporting's and leopard sightings from many years. The forest department sets trap camera and trap cages to capture these beasts. But the citizens need more for their safety. Leopards are known to live in hideouts and strike suddenly. Therefore, there is an urgent need for an alarm instantly when a leopard has been sighted. Today, fortunately most buildings and streets are under CCTV surveillance, keeping an eye on everything. The cameras are of high resolution with night vision to monitor every corner. Along with this, Security apps like MyGate, ADD ERP, ApnaComplex and more have replaced the traditional intercom. These Apps notifies the house owner as soon as the visitor passes the society's security. Only when the house owner approves the visit on the app, the visitor is allowed. The house owner can check and approve the visitors remotely from any location provided the user is connected to the internet. We propose a comprehensive object detection model that detects leopards from CCTV footages and integrates the findings with a Security application.

Using computer vision techniques for object detection, the model can identify and detect leopards. This provides incredible provision to citizens to detect leopard threats around them. Single Shot Detector, R-CNN (Regions with Convolutional Neural Networks) and Fast R-CNN, YOLO are some of the several object detection techniques that can be used [5]. In this paper we have used YOLO to detect leopards as its performance is better than many object detection algorithms. Since its debut, the YOLO algorithm has generated remarkable specifications. Its speed and accurateness are better than several top algorithms for object detection. Redmon et. al. [6] unveiled YOLO, an object detection model which can recognize multiple items in a photo while completing all object recognition phases in a single neural network. Through the use of bounding box coordinates & class prospects instead of image pixels, the object detection problem is reframed by YOLO as a single regression problem. The YOLO algorithm can be utilised for a variety of Computer Vision (CV) activities involving animals, Ariel images, the military, autonomous vehicles, sports, hospitals, and others etc. [7]

Over time YOLO has evolved from YOLOv1 to YOLOv8. For object detection in the leopard dataset, a deep learning approach is applied. It is able to acquire & retrieve the characteristics learned directly from the inputs, as opposed to prior machine learning methods that needed custom defining the features to be recovered from the inputs. The newest YOLOv8 algorithm is used in this research.

II. LITERATURE REVIEW

Automatic wildlife surveying and animal monitoring have become essential tools for conservation efforts and wildlife management. In recent years, CV techniques have played an important role in addressing the challenges associated with aerial video animal detection, especially in complex natural environments. This review of literature explores various studies and developments in the area of animal detection & segmentation by means of deep learning-based methods, with a focus on CNN and the popular "You Only Look Once" (YOLO) object detection algorithms.

Animal detection in aerial videos presents unique challenges due to the complexity of natural surroundings. One suggested method involves using global patterns of pixel motion, estimated through optical flow methods, to detect moving animals against the background [8]. This research has practical applications, such as observing locomotive behaviour to prevent animal disruptions in residential areas [9] [10].

The majority of deep learning-based animal detection systems are dominated by CNNs. Network's depth is determined by the number of layers in its architecture, as deep learning requires neural networks with numerous layers. CNNs are a type of feed-forward neural network that comprises three main layers: convolutional layers, pooling layers, and fully connected layers [11].

Norouzzadeh et al. conducted research using deep neural networks and achieved automatic animal detection with an

impressive accuracy of over 93.8% [12]. The key role of convolutional layers in this process is to create feature maps, acting as automatic feature extractors. Convolutional layer's output is downsampled by pooling layers. Finally, neurons from the input feature maps are joined to the internal neurons in the fully connected layer, enabling the network to make accurate predictions.

In another study by Saleh et al., the researchers frequently utilized transfer learning to create CNNs that are more accurate while using fewer resources [13]. It is a technique where a new categorization model is built using previously learned weights from a base model. This approach reduces training time and resource usage while achieving improved accuracy levels and requiring less data for training. Overall, CNNs have proven to be a potent tool for animal detection, & researchers have explored various approaches, together with transfer learning, to improve their efficiency and accuracy in detecting animals, including in challenging environments like urban and highway traffic scenarios.

In a study [14], transfer learning was employed using VGG-16 as the base model. VGG-16, with 33 stacked convolutional layers, achieved an impressive top-5 test accuracy of 92.7% on the ImageNet dataset, which consists of over 14 million images from 1000 classes. Maxpooling layers with 2x2 filters were used to reduce the volume size. The classification process involved a SoftMax classifier with 1000 channels for each class, followed by two fully linked layers, each with 4,096 nodes [11].

Meenatchi K et al [15] developed a Wild Animal Detection System using Artificial Intelligence on a Raspberry Pi. The system locally identifies wild animals in photographs. If a dangerous animal is detected, it employs a GSM module to transmit a message and emits an ultrasonic buzzer to scare away the animal. This comprehensive solution aims to mitigate harm to both people and animals, as well as protect valuable resources, through the deployment of a deep convolutional neural network-based animal detection system.

In their research on "Real-time Animal Detection and Prevention System for Crop Fields," R Lathesparan et al [16] utilized two Convolutional Neural Network (CNN) classification models combined and tested on a Raspberry Pi as the processing unit. The system captures pictures of animals using a thermal sensor on Arduino, which triggers a picture whenever an animal is detected. The researchers incorporated sudden light flashes, ultrasonography, and bee sounds to scare away the animals. The categorization model achieved an accuracy rate of 77 percent.

Verma et al. [24] worked with a camera-trap database, utilizing candidate animal proposals to develop a verification step that determines whether a specific patch in the image is background or an animal. Their results on a conventional camera-trap dataset revealed an accuracy of 91.4%, showcasing the effectiveness of their approach in animal detection.

Dhillon et al. [25] proposed a deep learning-based system for detecting wild animals from highly cluttered natural forest images. Their detection method using Residual Network (ResNet) outperformed existing systems. In the context of deep learning techniques for animal recognition, segmentation, and detection, a survey was conducted to provide a concise analysis and comparison of various approaches [26].

In another study [27], the focus was on animal object detection and segmentation in wildlife monitoring videos captured by camera-traps. The experimental results demonstrated that their framework achieved superior animal object detection accuracy, surpassing state-of-the-art approaches, including Faster-RCNN, by up to 4.5%.

The rapid advancement of computer vision has led to neural networks dominating object detection techniques. Deep learning-based methods initially started with text classification and then evolved to identify human behaviour, ranging from simple tasks to complex group activities. The YOLO object identification algorithms, particularly between versions 1 and 8, have outperformed traditional algorithms in terms of capability and performance.

Addressing the challenge of face detection in complex and real-world environments, a customized version of the YOLOv3 object detector was proposed in a paper [28]. This specialized model was designed for accurate and fast face detection in surveillance and biometrics scenarios.

Additionally, researchers in the field of industrial automation focused on automating the monitoring, categorization, and segregation of industrial gears in assembly lines. Three object identification models—Faster RCNN, YOLO, and SSD—were evaluated, and YOLOv4 was chosen as the preferred model due to its optimal trade-off between accuracy and detection speed [30].

A novel variation of the YOLO family called YOLOv5 was introduced by a researcher named Glenn and his team. This version, as highlighted by Nepal and Eslamiat [17], utilizes PyTorch instead of Darknet and incorporates CSPDarknet53 as its structural support. YOLOv5 resolves the issue of repetitive gradient information found in YOLOv3 and YOLOv4.

Another breakthrough in real-time object detection is YOLOv7, which has been revolutionizing the computer vision field. This model was trained using only the MS COCO dataset [18]. YOLOv7 offers remarkable features and improvements, making it ground-breaking in the computer vision industry.

The YOLOv7 model has shown extraordinary progresses over its former versions on Graphics Processing Units (GPU) V100. It outperforms all former models by attaining a speed of 30 FPS or more while sustaining high accuracy levels. YOLOv7 has achieved the highest mean Average Precision (mAP) of 56.8%. Notably, YOLOv7 succeeded to reduce approximately 40% of parameters and 50% of computation, leading to amplified accuracy and decreased inference cost in real-time

object detection. The model also demonstrated faster inference speed and better detection accuracy.

Terven et al. [19] provided an overview of the YOLO series, starting from its commencement and covering its evolution until YOLOv8, which was released in January 2023 by Ultralytics. YOLOv8 has made some notable improvements, including eliminating anchors, resulting in fewer box predictions and faster Non-maximum Suppression (NMS). During training, YOLOv8 utilizes mosaic augmentation, but this augmentation is disabled for the final ten epochs due to potential harmful effects when used continuously. YOLOv8 can be installed as a PIP package or executed through the Command Line Interface. The model offers five scaled variants: YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x.

Patel et al. [20] conducted a study on object detection in hockey sports using YOLOv3, achieving a high accuracy of 91.3%. The model demonstrated the potential of YOLOv3 in sports analytics, especially in the detection and tracking of objects in dynamic environments like hockey games.

Another advancement in object detection is the defect detection model built on YOLOv5 by Yao et al. in 2021. This model excels in accurately and swiftly detecting faults. To enhance its ability to identify minute flaws, a small object identification layer was added. Additionally, the model incorporated the layer of squeeze-and-excitation and full Intersection over Union (IoU) of the loss function to improve regression accuracy. The Cosine Annealing algorithm was employed to enhance the model's performance after being trained using transfer learning. YOLOv5 achieved an mAP@0.5 of 94.7%, representing an approximate 9% increase compared to the original method.

Liu et al. [21] conducted a study on railway signal lights detection and found YOLOv5 to be highly useful in their experiment. The model was trained on a subway scenes dataset with signal lights and achieved a remarkable running speed of 100 FPS. Furthermore, the model exhibited an average recall rate and accuracy of 97.2%, highlighting its efficacy in detecting and recognizing railway signal lights for safety and operational purposes.

Hao et al. [22] proposed a lightweight method to enhance the speed and accuracy of YOLOv5. Their modified model, called Light-YOLOv5, was tested on a dataset that included fire scenario examples. The results demonstrated that Light-YOLOv5 achieved an impressive Frames Per Second (FPS) of 91.1 and increased the mAP by 0.033. Notably, the mAP of the modified model was 6.8% higher than YOLOv7-tiny, showcasing the efficiency of the algorithm in real-time object detection tasks.

To address operational lag, inspection and certification expenses, and undetected harm from human error, Hussain et al. [23] introduced a framework built using YOLOv7. The authors suggested a domain variance modelling approach to overcome the problem of data scarcity by generating representative data

Dhillon et al. [25] proposed a deep learning-based system for detecting wild animals from highly cluttered natural forest images. Their detection method using Residual Network (ResNet) outperformed existing systems. In the context of deep learning techniques for animal recognition, segmentation, and detection, a survey was conducted to provide a concise analysis and comparison of various approaches [26].

In another study [27], the focus was on animal object detection and segmentation in wildlife monitoring videos captured by camera-traps. The experimental results demonstrated that their framework achieved superior animal object detection accuracy, surpassing state-of-the-art approaches, including Faster-RCNN, by up to 4.5%.

The rapid advancement of computer vision has led to neural networks dominating object detection techniques. Deep learning-based methods initially started with text classification and then evolved to identify human behaviour, ranging from simple tasks to complex group activities. The YOLO object identification algorithms, particularly between versions 1 and 8, have outperformed traditional algorithms in terms of capability and performance.

Addressing the challenge of face detection in complex and real-world environments, a customized version of the YOLOv3 object detector was proposed in a paper [28]. This specialized model was designed for accurate and fast face detection in surveillance and biometrics scenarios.

Additionally, researchers in the field of industrial automation focused on automating the monitoring, categorization, and segregation of industrial gears in assembly lines. Three object identification models—Faster RCNN, YOLO, and SSD—were evaluated, and YOLOv4 was chosen as the preferred model due to its optimal trade-off between accuracy and detection speed [30].

A novel variation of the YOLO family called YOLOv5 was introduced by a researcher named Glenn and his team. This version, as highlighted by Nepal and Eslamiat [17], utilizes PyTorch instead of Darknet and incorporates CSPDarknet53 as its structural support. YOLOv5 resolves the issue of repetitive gradient information found in YOLOv3 and YOLOv4.

Another breakthrough in real-time object detection is YOLOv7, which has been revolutionizing the computer vision field. This model was trained using only the MS COCO dataset [18]. YOLOv7 offers remarkable features and improvements, making it ground-breaking in the computer vision industry.

The YOLOv7 model has shown extraordinary progresses over its former versions on Graphics Processing Units (GPU) V100. It outperforms all former models by attaining a speed of 30 FPS or more while sustaining high accuracy levels. YOLOv7 has achieved the highest mean Average Precision (mAP) of 56.8%. Notably, YOLOv7 succeeded to reduce approximately 40% of parameters and 50% of computation, leading to amplified accuracy and decreased inference cost in real-time

object detection. The model also demonstrated faster inference speed and better detection accuracy.

Terven et al. [19] provided an overview of the YOLO series, starting from its commencement and covering its evolution until YOLOv8, which was released in January 2023 by Ultralytics. YOLOv8 has made some notable improvements, including eliminating anchors, resulting in fewer box predictions and faster Non-maximum Suppression (NMS). During training, YOLOv8 utilizes mosaic augmentation, but this augmentation is disabled for the final ten epochs due to potential harmful effects when used continuously. YOLOv8 can be installed as a PIP package or executed through the Command Line Interface. The model offers five scaled variants: YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x.

Patel et al. [20] conducted a study on object detection in hockey sports using YOLOv3, achieving a high accuracy of 91.3%. The model demonstrated the potential of YOLOv3 in sports analytics, especially in the detection and tracking of objects in dynamic environments like hockey games.

Another advancement in object detection is the defect detection model built on YOLOv5 by Yao et al. in 2021. This model excels in accurately and swiftly detecting faults. To enhance its ability to identify minute flaws, a small object identification layer was added. Additionally, the model incorporated the layer of squeeze-and-excitation and full Intersection over Union (IoU) of the loss function to improve regression accuracy. The Cosine Annealing algorithm was employed to enhance the model's performance after being trained using transfer learning. YOLOv5 achieved an mAP@0.5 of 94.7%, representing an approximate 9% increase compared to the original method.

Liu et al. [21] conducted a study on railway signal lights detection and found YOLOv5 to be highly useful in their experiment. The model was trained on a subway scenes dataset with signal lights and achieved a remarkable running speed of 100 FPS. Furthermore, the model exhibited an average recall rate and accuracy of 97.2%, highlighting its efficacy in detecting and recognizing railway signal lights for safety and operational purposes.

Hao et al. [22] proposed a lightweight method to enhance the speed and accuracy of YOLOv5. Their modified model, called Light-YOLOv5, was tested on a dataset that included fire scenario examples. The results demonstrated that Light-YOLOv5 achieved an impressive Frames Per Second (FPS) of 91.1 and increased the mAP by 0.033. Notably, the mAP of the modified model was 6.8% higher than YOLOv7-tiny, showcasing the efficiency of the algorithm in real-time object detection tasks.

To address operational lag, inspection and certification expenses, and undetected harm from human error, Hussain et al. [23] introduced a framework built using YOLOv7. The authors suggested a domain variance modelling approach to overcome the problem of data scarcity by generating representative data

D. Performance metrics

Using the created dataset Precision, Recall, and Average Precision (AP) were computed and contrasted to assess model's performance. The following formulae were used to compute the metrics.

Precision(P): By dividing the number of correctly identified leopards by the total number of leopards discovered, the classification accuracy is determined. It can be written as

$$Precision = \frac{True\ Positive}{(True\ Positive + True\ Negative)}$$

Recall(R): In a dataset, it is the ratio of precisely observed leopards to the total number of leopards.

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

F1-score: It is used to achieve a balance between precision and recall. The harmonic of recall and precision is how the F1-score is represented. Compared to accuracy metrics, it is considered to be a more accurate measure.

$$F1\ score = \left(\frac{Recall^{-1} + Precision^{-1}}{2} \right)^{-1}$$

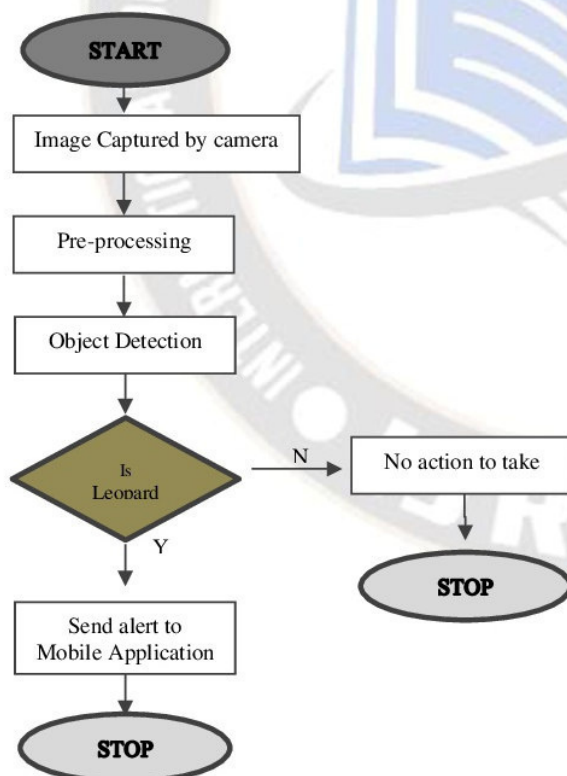


Figure 1. Proposed Architecture

True positives are positive samples classified correctly, false positives are negative samples classified incorrectly, and false negatives are positive samples classified incorrectly.

MAP and AP: The average precision for each class is known as MAP. At different thresholds, the AP provides total weighted precision. Either the mean average of all classes or the sum of all IOU criteria are used to determine the MAP value.

FPS:- Number of images the model processes per second is frames per second i.e. FPS. It is a very important speed performance parameter.

IV. RESULTS

The experiment involved training the YOLOv8 model on different images of leopards. Table I displays the parameters used during the training of the neural network.

The gradient descent optimization algorithm plays a crucial role in adjusting the model's weights in relation to the loss function. The learning rate parameter determines how much the model's weights are modified in each iteration. A higher learning rate may result in faster convergence, but it could also lead to overshooting the optimal solution. On the other hand, a lower learning rate may lead to slow convergence. Finding an appropriate learning rate is essential for optimizing the model's performance.

Stochastic Gradient Descent (SGD) is employed as the optimization algorithm. It accumulates the gradients from earlier stages and determines the direction in which the weights should be adjusted. This allows the algorithm to efficiently optimize the model's parameters.

Due to the large volume of data used in the experiment, the concept of batch size is utilized. The batch size refers to the number of samples the algorithm uses in each iteration to train the network. By processing data in batches, the training process becomes handier and technically efficient.

The number of epochs represents the total number of times the entire dataset undergoes training. Each epoch involves passing the entire dataset through the network. Training for multiple epochs allows the model to learn from the data multiple times, which can improve its ability to generalize to unseen examples.

In YOLOv8, lr0 and lrf are used to control the learning rate schedule during training. lr0 denotes the initial learning rate, and lrf represents the final learning rate at the last epoch of training. By using a learning rate schedule, the model can adaptively adjust the learning rate during training, which may lead to better convergence and overall performance.

TABLE I. TRAINING PARAMETERS ON YOLO FOR LEOPARD DETECTION

Parameters	Value
<i>lr0</i>	0.01
<i>Lrf</i>	0.01
<i>Momentum</i>	0.937
<i>Weight decay</i>	0.0010078125
<i>Optimizer</i>	SGD (<i>lr</i> =0.01)
<i>Epochs</i>	100
<i>Batch size</i>	16
<i>Image size</i>	640

The default values for both *lr0* and *lrf* in YOLOv8 are set to 0.01. During training, this value is used to regulate the learning rate. Fine-tuning these parameters and experimenting with different learning rate schedules can help optimize the model's performance in leopard threat detection.

Overall, the experiment's results and the discussion of training parameters provide valuable insights into the effectiveness and efficiency of the proposed system in alerting users of leopard threat detection through the security application.

Table II displays the experimental result of our model on our dataset. It was determined that 100 epochs would be needed to train the model. The model that was trained showed promising performance.

TABLE II. RESULTS OF YOLOV8 MODEL

<i>Precision</i>	0.951
<i>Recall</i>	0.915
<i>F1 Score</i>	0.932
<i>mAP@0.5</i>	0.959
<i>mAP@0.5:.95</i>	0.597
<i>Weight</i>	136.7MB
<i>No. of Parameters</i>	68124531

The *mAP@0.5* value of YOLOv5 is 0.959 with a precision of 0.951 & a recall of 0.915. The Precision-Confidence curve, Recall -Confidence curve & Confusion matrix is displayed in figure 2,3 & 4 respectively.

Mean Average Precision (mAP) and detection speed are the criteria taken into consideration while assessing the suggested model's performance.

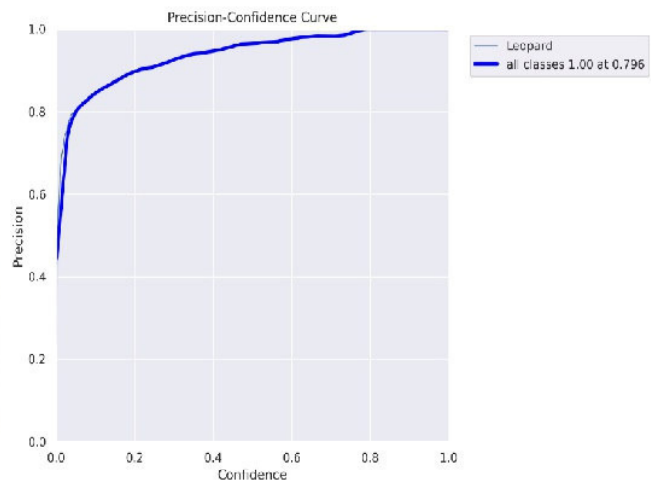


Figure 2- Precision-Confidence Curve

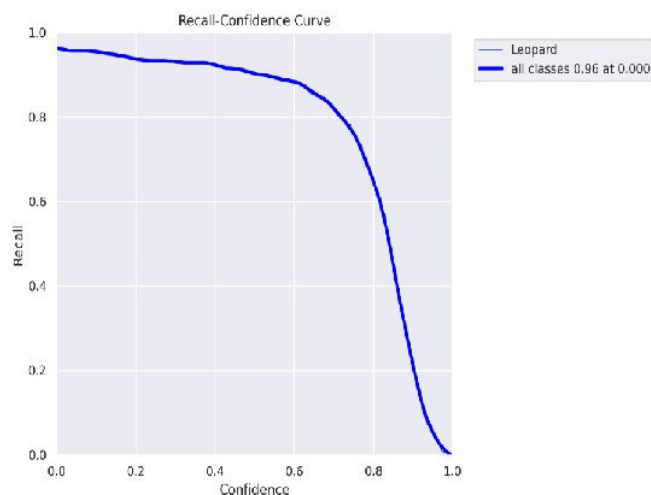


Figure 3- Recall-Confidence Curve

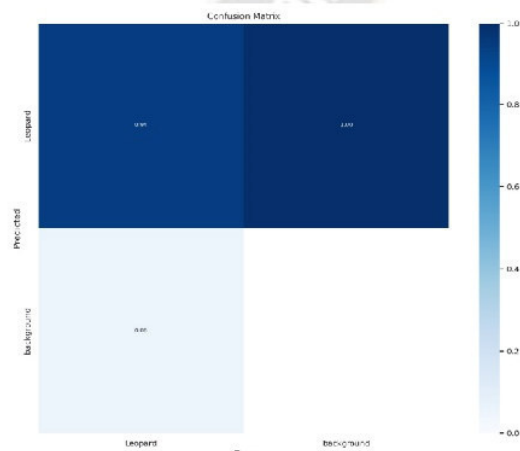


Figure 4- Confusion matrix

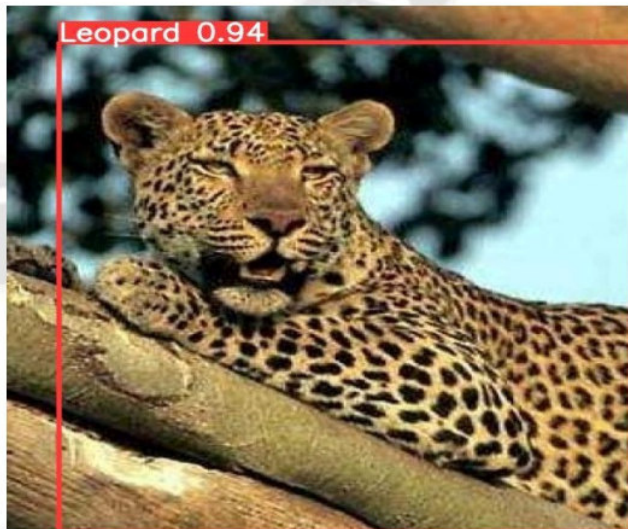
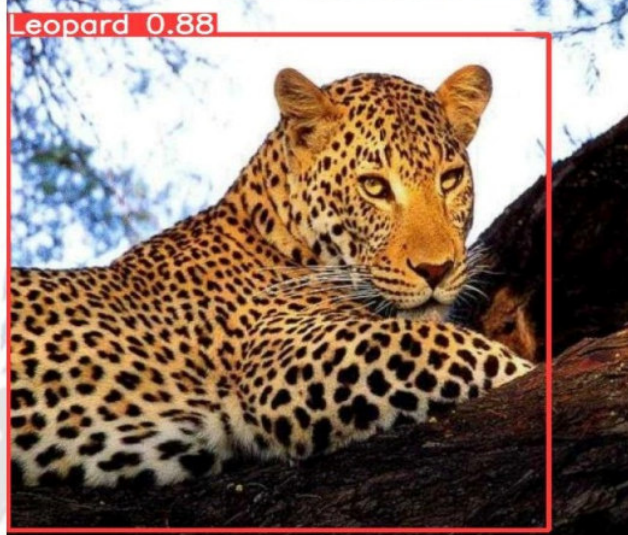
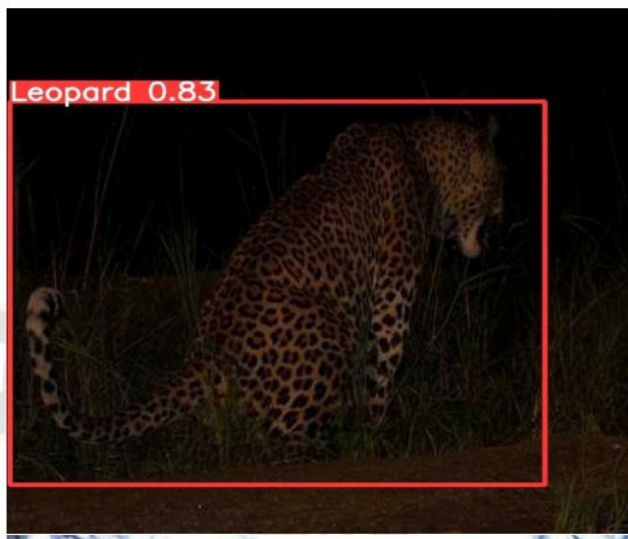




Figure 5- Testing results of the proposed system

The mAP and detection speed of the proposed method are 95.9% at 37 fps. The proposed model has achieved an accuracy of 0.959. It is found that the YOLOv8 model performs good for all the performance metrics. The Testing results of the proposed system are displayed in figure 5.

V. CONCLUSION

This paper presents the most recent version of YOLO for detecting leopards in residential areas and cautioning users via a security application. The model has been precisely intended to attain high accuracy, even during night-time. The results indicate that the model's accuracy in leopard detection exceeds 95%, which is highly effective for the proposed application.

In addition to accuracy, the inference speed of the algorithm was also a crucial metric considered in the study. The model demonstrated a notable inference speed of more than 35 frames

per second (FPS), allowing for real-time detection and rapid alerts to ensure timely responses to potential leopard sightings.

It is noteworthy that while there are some examples of deep learning algorithms being applied to leopard detection in existing works, this paper represents a revolutionary effort. To the best of the authors' knowledge, this is the first of its kind study to influence deep learning techniques for automated leopard detection and providing real-time information to people. This innovative application of deep learning technology marks a critical need in enhancing wildlife safety and justifying potential risks to humans and animals in populated areas.

Moreover, the paper intends to contribute to the field of wildlife detection using deep learning and inspires further research and application of similar algorithms for detecting other wild species beyond leopards.

The successful implementation and high accuracy of the proposed model for leopard detection open up possibilities for its application in other contexts and scenarios. By sharing their findings, the authors hope to inspire further studies and innovations in the realm of wildlife conservation and safety using cutting-edge deep learning algorithms.

This research presents an Automated Leopard Alert and Reporting Mechanism that addresses the rising concern of leopard sightings and potential attacks in metropolitan cities like Mumbai. The rapid urban development and destruction of natural environments have forced leopards and other wildlife to move closer to human settlements, posing a threat to public safety.

The proposed solution utilizes deep learning-based object detection, employing the YOLO algorithm, to detect leopards from CCTV footage. The model achieved a notable accuracy of 95.9% and a high detection speed of 37 frames per second. By integrating these detection capabilities with existing security applications and surveillance systems, real-time alerts can be provided to citizens about leopard presence in their vicinity. This sanctions residents to take necessary defenses and authorities to respond promptly, minimizing potential conflicts and ensuring the safety of both humans and leopards.

The application of computer vision techniques and deep learning models, mainly YOLO, exhibits the potential of object detection in addressing wildlife threats. The elasticity of object detection technology encompasses beyond leopard detection, offering applications in multiple arenas, including aerial image analysis, military operations, self-governing vehicles, and more.

The Automated Leopard Alert and Reporting Mechanism presents a practical and hands-on approach concerning leopard sightings in residential areas. By using deep learning and along with existing smart security systems, it offers an effective solution for safeguarding public safety.

Future work demands expanding the dataset used for training, combining a vivid and challenging image conditions,

such as rain and hostile weather. Training the model on a larger and more wide-ranging dataset will improve its robustness in real-world scenarios. Moreover, continuous research and implementation of automated alert systems can contribute to fostering harmonious coexistence between humans and wildlife in the face of urbanization.

In conclusion, the Automated Leopard Alert and Reporting Mechanism provides a valuable contribution to wildlife conservation efforts and public safety in urban areas. By utilizing advanced technologies and embracing further research, we can continue to develop innovative solutions that protect both human communities and wildlife in rapidly evolving urban landscapes.

REFERENCES

- [1] <https://timesofindia.indiatimes.com/city/mumbai/sgnp-census-confirms-41-leopards-27-of-them-new/articleshow/62812269.cms>
- [2] <https://timesofindia.indiatimes.com/city/mumbai/mumbai-woman-hospitalised-after-leopard-attacks-her-at-aarey-colony/articleshow/95459869.cms?from=mdr>
- [3] <https://www.mid-day.com/mumbai/mumbai-news/article/mumbai-leopard-attacks-38-year-old-woman-in-aarey-colony-23255164>
- [4] <https://www.timesnownews.com/the-buzz/article/video-leopard-attacks-two-stray-dogs-in-mumbai-watch-chilling-footage/525364>
- [5] Padilla, R., Passos, W. L., Dias, T. L., Netto, S. L., & da Silva, E. A. (2021). A comparative analysis of object detection metrics with a companion open-source toolkit. *Electronics*, 10(3), 279.
- [6] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 779-788).
- [7] Gorriz, J. M., Ramirez, J., Ortiz, A., Martinez-Murcia, F. J., Segovia, F., Suckling, J. & Ferrandez, J. M. (2020). Artificial intelligence within the interplay between natural and artificial computation: Advances in data science, trends and applications. *Neurocomputing*, 410, 237-270.
- [8] Nguyen, H., MacLagan, S. J., Nguyen, T. D., Nguyen, T., Flemons, P., Andrews, K., ... & Phung, D. (2017, October). Animal recognition and identification with deep convolution neural networks for automated Animal monitoring. In *Data Science and Advanced Analytics (DSAA), 2017 IEEE International Conference on* (pp. 40- 49). IEEE.
- [9] Kumar, S., & Singh, S. K. (2016). Monitoring of pet animal in smart cities using animal biometrics. *Future Generation Computer Systems*.
- [10] Xue, C., Wang, P., Zhao, J., Xu, A., & Guan, F. (2017). Development and validation of a universal primer pair for the simultaneous detection of eight animal species. *Food chemistry*, 221, 790-796.
- [11] Karen Simonyan, A. Z. (2015) 'very deep convolutional networks for large-scale image recognition', pp. 1-14.
- [12] Norouzzadeh, M. S. et al. (2017) 'Automatically identifying, counting, and describing wild animals in camera-trap images with deep learning', pp. 1- 17. Available at: <http://arxiv.org/abs/1703.05830>.
- [13] Saleh, K., Hossny, M. and Nahavandi, S. (2018) 'Effective vehicle-based kangaroo detection for collision warning systems using region-based convolutional networks', *Sensors (Switzerland)*, 18(6). doi: 10.3390/s18061913.
- [14] Willi, M. et al. (2019) 'Identifying animal species in camera trap images using deep learning and citizen science', *Methods in Ecology and Evolution*, 10(1), pp. 80-91. doi: 10.1111/2041210X.13099.
- [15] Wild Animal Detection System Using Deep Convolutional Neural Networks Meenatchi K1 , Thibishini V International Journal of Scientific Research in Science and Technology Print ISSN: 2395-6011 | Online ISSN: 2395-602X (www.ijrst.com) doi : <https://doi.org/10.32628/IJSRST.1.Vaisnavi.K1.Mrs.R.Ahila.2>
- [16] Real-time Animal Detection and Prevention System for Crop Fields R Lathesparan#, A Sharanjah, R Thushanth, S Kenurshan, MNM Nifras, and WU Wickramaarachi, 13th International Research Conference General Sir John Kotelawala Defence University
- [17] Nepal, U., & Eslamiat, H. (2022). Comparing YOLOv3, YOLOv4 and YOLOv5 for Autonomous Landing Spot Detection in Faulty UAVs. *Sensors*, 22(2), 464
- [18] Wang, C. Y., Bochkovskiy, A., & Liao, H. Y. M. (2022). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv preprint arXiv:2207.02696*.
- [19] Terven et al.(2023). A comprehensive review of yolo: from yolov1 to Yolov8 and beyond arXiv:2304.00501v1
- [20] Patel et al(2023).Object detection in hockey sport video via pretrained yolov3 based deep learning model. *ICTACT Journal on Image and Video Processing*, February 2023, Volume: 13, Issue: 03
- [21] Liu, W., Wang, Z., Zhou, B., Yang, S., & Gong, Z. (2021, May). Real-time signal light detection based on yolov5 for railway. In *IOP Conference Series: Earth and Environmental Science* (Vol. 769, No. 4, p. 042069). IOP Publishing.
- [22] Hao, X., Bo, L., & Fei, Z. (2021). Light-YOLOv5: A Lightweight Algorithm for Improved YOLOv5 in Complex Fire Scenarios.
- [23] Hussain, M., Al-Aqrabi, H., Munawar, M., Hill, R., & Alsaboui, T., (2022). Domain Feature Mapping with YOLOv7 for Automated Edge-Based Pallet Racking Inspections. *Sensors*, 22, 6927.
- [24] Verma, G.K., Gupta, P. (2018). Wild Animal Detection Using Deep Convolutional Neural Network. In: Chaudhuri, B., Kankanhalli, M., Raman, B. (eds) *Proceedings of 2nd International Conference on Computer Vision & Image Processing*. Advances in Intelligent Systems and Computing, vol 704. Springer, Singapore. https://doi.org/10.1007/978-981-10-7898-9_27
- [25] Dhillon, A., Verma, G.K. (2018). Wild Animal Detection from Highly Cluttered Forest Images Using Deep Residual Networks. In: Tiwary, U. (eds) *Intelligent Human Computer Interaction. IHCI 2018. Lecture Notes in Computer Science()*, vol 11278. Springer, Cham. https://doi.org/10.1007/978-3-030-04021-5_21
- [26] V. Palanisamy and N. Ratnarajah, "Detection of Wildlife Animals using Deep Learning Approaches: A Systematic Review," 2021 21st International Conference on Advances in ICT for Emerging Regions (ICter), Colombo, Sri Lanka, 2021, pp. 153-158, doi: 10.1109/ICter53630.2021.9774826.
- [27] Z. Zhang, Z. He, G. Cao and W. Cao, "Animal Detection From Highly Cluttered Natural Scenes Using Spatiotemporal Object Region Proposals and Patch Verification," in *IEEE*

such as rain and hostile weather. Training the model on a larger and more wide-ranging dataset will improve its robustness in real-world scenarios. Moreover, continuous research and implementation of automated alert systems can contribute to fostering harmonious coexistence between humans and wildlife in the face of urbanization.

In conclusion, the Automated Leopard Alert and Reporting Mechanism provides a valuable contribution to wildlife conservation efforts and public safety in urban areas. By utilizing advanced technologies and embracing further research, we can continue to develop innovative solutions that protect both human communities and wildlife in rapidly evolving urban landscapes.

REFERENCES

- [1] <https://timesofindia.indiatimes.com/city/mumbai/sgnp-census-confirms-41-leopards-27-of-them-new/articleshow/62812269.cms>
- [2] <https://timesofindia.indiatimes.com/city/mumbai/mumbai-woman-hospitalised-after-leopard-attacks-her-at-aarey-colony/articleshow/95459869.cms?from=mdr>
- [3] <https://www.mid-day.com/mumbai/mumbai-news/article/mumbai-leopard-attacks-38-year-old-woman-in-aarey-colony-23255164>
- [4] <https://www.timesnownews.com/the-buzz/article/video-leopard-attacks-two-stray-dogs-in-mumbai-watch-chilling-footage/525364>
- [5] Padilla, R., Passos, W. L., Dias, T. L., Netto, S. L., & da Silva, E. A. (2021). A comparative analysis of object detection metrics with a companion open-source toolkit. *Electronics*, 10(3), 279.
- [6] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 779-788).
- [7] Gorriz, J. M., Ramirez, J., Ortiz, A., Martinez-Murcia, F. J., Segovia, F., Suckling, J., & Ferrandez, J. M. (2020). Artificial intelligence within the interplay between natural and artificial computation: Advances in data science, trends and applications. *Neurocomputing*, 410, 237-270.
- [8] Nguyen, H., Maclagan, S. J., Nguyen, T. D., Nguyen, T., Flemons, P., Andrews, K., ... & Phung, D. (2017, October). Animal recognition and identification with deep convolution neural networks for automated Animal monitoring. In *Data Science and Advanced Analytics (DSAA)*, 2017 IEEE International Conference on (pp. 40- 49). IEEE.
- [9] Kumar, S., & Singh, S. K. (2016). Monitoring of pet animal in smart cities using animal biometrics. *Future Generation Computer Systems*.
- [10] Xue, C., Wang, P., Zhao, J., Xu, A., & Guan, F. (2017). Development and validation of a universal primer pair for the simultaneous detection of eight animal species. *Food chemistry*, 221, 790-796.
- [11] Karen Simonyan, A. Z. (2015) 'very deep convolutional networks for large-scale image recognition', pp. 1-14.
- [12] Norouzzadeh, M. S. et al. (2017) 'Automatically identifying, counting, and describing wild animals in camera-trap images with deep learning', pp. 1- 17. Available at: <http://arxiv.org/abs/1703.05830>.
- [13] Saleh, K., Hossny, M. and Nahavandi, S. (2018) 'Effective vehicle-based kangaroo detection for collision warning systems using region-based convolutional networks', *Sensors (Switzerland)*, 18(6). doi: 10.3390/s18061913.
- [14] Willi, M. et al. (2019) 'Identifying animal species in camera trap images using deep learning and citizen science', *Methods in Ecology and Evolution*, 10(1), pp. 80-91. doi: 10.1111/2041210X.13099.
- [15] Wild Animal Detection System Using Deep Convolutional Neural Networks Meenatchi K1 , Thibishini V International Journal of Scientific Research in Science and Technology Print ISSN: 2395-6011 | Online ISSN: 2395-602X (www.ijrst.com) doi : <https://doi.org/10.32628/IJSRST.1.Vaisnavi.K1.Mrs.R.Ahila.2>
- [16] Real-time Animal Detection and Prevention System for Crop Fields R Lathesparan#, A Sharanjah, R Thushanth, S Kenurshan, MNM Nifras, and WU Wickramaarachi, 13th International Research Conference General Sir John Kotelawala Defence University
- [17] Nepal, U., & Eslamiat, H. (2022). Comparing YOLOv3, YOLOv4 and YOLOv5 for Autonomous Landing Spot Detection in Faulty UAVs. *Sensors*, 22(2), 464
- [18] Wang, C. Y., Bochkovskiy, A., & Liao, H. Y. M. (2022). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv preprint arXiv:2207.02696*.
- [19] Terven et al.(2023). A comprehensive review of yolo: from yolov1 to Yolov8 and beyond arXiv:2304.00501v1
- [20] Patel et al(2023).Object detection in hockey sport video via pretrained yolov3 based deep learning model. *ICTACT Journal on Image and Video Processing*, February 2023, Volume: 13, Issue: 03
- [21] Liu, W., Wang, Z., Zhou, B., Yang, S., & Gong, Z. (2021, May). Real-time signal light detection based on yolov5 for railway. In *IOP Conference Series: Earth and Environmental Science* (Vol. 769, No. 4, p. 042069). IOP Publishing.
- [22] Hao, X., Bo, L., & Fei, Z. (2021). Light-YOLOv5: A Lightweight Algorithm for Improved YOLOv5 in Complex Fire Scenarios.
- [23] Hussain, M., Al-Aqrabi, H., Munawar, M., Hill, R., & Alsboui, T., (2022). Domain Feature Mapping with YOLOv7 for Automated Edge-Based Pallet Racking Inspections. *Sensors*, 22, 6927.
- [24] Verma, G.K., Gupta, P. (2018). Wild Animal Detection Using Deep Convolutional Neural Network. In: Chaudhuri, B., Kankanhalli, M., Raman, B. (eds) *Proceedings of 2nd International Conference on Computer Vision & Image Processing*. Advances in Intelligent Systems and Computing, vol 704. Springer, Singapore. https://doi.org/10.1007/978-981-10-7898-9_27
- [25] Dhillon, A., Verma, G.K. (2018). Wild Animal Detection from Highly Cluttered Forest Images Using Deep Residual Networks. In: Tiwary, U. (eds) *Intelligent Human Computer Interaction. IHCI 2018. Lecture Notes in Computer Science()*, vol 11278. Springer, Cham. https://doi.org/10.1007/978-3-030-04021-5_21
- [26] V. Palanisamy and N. Ratnarajah, "Detection of Wildlife Animals using Deep Learning Approaches: A Systematic Review," 2021 21st International Conference on Advances in ICT for Emerging Regions (ICTer), Colombo, Sri Lanka, 2021, pp. 153-158, doi: 10.1109/ICTer53630.2021.9774826.
- [27] Z. Zhang, Z. He, G. Cao and W. Cao, "Animal Detection From Highly Cluttered Natural Scenes Using Spatiotemporal Object Region Proposals and Patch Verification," in IEEE

CERTIFICATES





T. Z. A. S. P. MANDAL'S
PRAGATI COLLEGE OF ARTS & COMMERCE,
DOMBIVLI (E)

AFFILIATED TO UNIVERSITY OF MUMBAI
2(f) & 12(b) Status by UGC,
Re-accredited with B++ grade by NAAC

Certificate of Participation

This is to certify that **MISS. NEHA VORA** of **PACIFIC ACADEMY OF HIGHER EDUCATION AND RESEARCH UNIVERSITY, UDAIPUR, RAJASTHAN, INDIA** Participated in One-Day National Level Multidisciplinary E-Conference on **DIGITAL TRANSFORMATION: NAVIGATING THE NEW FRONTIER** organized by Department of Self-Financing Courses on 15th February, 2024. She Presented a paper titled **"COMPARATIVE ANALYSIS OF YOLOV5, YOLOV7 & YOLOV8 IN SPORTS OBJECT DETECTION"**.

Ms. Sneha Mhatre
(Convener)

Dr. Jyoti Pohane
(Principal)



VELAMMAL
INSTITUTE OF TECHNOLOGY

Approved by AICTE New Delhi & Anna University Chennai
Velammal Knowledge Park, Chennai - Kolkatta Highway, Ponneri 601204



ICA5NT 2023

Certificate of Appreciation

is hereby granted to

Mrs.Neha Vora

Pacific Academy Of Higher Education And Research University, Udaipur, Rajasthan

for participation and presentation of paper entitled

Comparative Study Of Target Image Detection Using Deep Learning

which was presented at Third International Conference on Artificial Intelligence, 5G Communications and Network Technologies (ICA5NT 2023) organized by the **Department of Electronics and Communication Engineering**, Velammal Institute of Technology, Chennai on 23rd & 24th March 2023.

Coordinator
Dr.R.Jothi Chitra
Professor-ECE

Coordinator
Dr.M.Sivarathinabala
Associate Professor-ECE

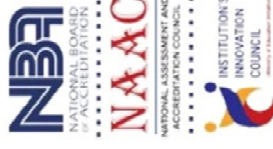
Convener
Dr.B.Sridevi
Professor & Head-ECE

Principal
Dr.N.Balaji



VELAMMAL
INSTITUTE OF TECHNOLOGY

Approved by AICTE New Delhi & Anna University Chennai
Velammal Knowledge Park, Chennai - Kolkatta Highway, Ponneri 601204



ICA5NT 2023

Certificate of Appreciation

is hereby granted to

Mrs.Neha Vora

Pacific Academy Of Higher Education And Research University, Udaipur, Rajasthan

for participation and presentation of paper entitled

Comparative Study Of Target Image Detection Using Deep Learning

which was presented at Third International Conference on Artificial Intelligence, 5G Communications and Network Technologies (ICA5NT 2023) organized by the **Department of Electronics and Communication Engineering**, Velammal Institute of Technology, Chennai on 23rd & 24th March 2023.

Coordinator
Dr.R.Jothi Chitra
Professor-ECE

Coordinator
Dr. M.Sivarathinabala
Associate Professor-ECE

Convener
Dr.B.Sridevi
Professor & Head-ECE

Principal
Dr.N.Balaji

PLAGIARISM REPORT



Intelligent Homicide Investigator: A Unique Homicide Crime Scene Investigation and Data Collection Tool using Convolutional Neural Network

by NEHA VORA

Submission date: 30-Aug-2024 12:04PM (UTC+0530)

Submission ID: 2441051767

File name: complete_thesis.pdf (5.59M)

Word count: 33382

Character count: 188720

Intelligent Homicide Investigator: A Unique Homicide Crime Scene Investigation and Data Collection Tool using Convolutional Neural Network

ORIGINALITY REPORT

6%

SIMILARITY INDEX

4%

INTERNET SOURCES

5%

PUBLICATIONS

2%

STUDENT PAPERS

PRIMARY SOURCES

1

www.ijnrd.org

Internet Source

1%

2

link.springer.com

Internet Source

1%

3

Jianyu Xiao, Shancang Li, Qingliang Xu.
"Video-Based Evidence Analysis and
Extraction in Digital Forensic Investigation",
IEEE Access, 2019

Publication

1%

4

www.ijarse.com

Internet Source

<1%

5

Rakhsith L. A, Anusha K. S, Karthik B. E, Arun
Nithish D, Kishore Kumar V. "A Survey on
Object Detection Methods in Deep Learning",
2021 Second International Conference on
Electronics and Sustainable Communication
Systems (ICESC), 2021

Publication

<1%

6	docplayer.net Internet Source	<1 %
7	www.sersc.org Internet Source	<1 %
8	Jun Deng, Xiaojing Xuan, Weifeng Wang, Zhao Li, Hanwen Yao, Zhiqiang Wang. "A review of research on object detection based on deep learning", Journal of Physics: Conference Series, 2020 Publication	<1 %
9	technodocbox.com Internet Source	<1 %
10	digital.lib.washington.edu Internet Source	<1 %
11	kentuckystatepolice.org Internet Source	<1 %
12	www.ijraset.com Internet Source	<1 %
13	www.ijsdr.org Internet Source	<1 %
14	Submitted to University Politehnica of Bucharest Student Paper	<1 %
15	"Intelligent Computing Methodologies", Springer Science and Business Media LLC,	<1 %

2022

Publication

16

Submitted to CITY College, Affiliated Institute of the University of Sheffield

Student Paper

<1 %

17

Zhengxia Zou, Keyan Chen, Zhenwei Shi, Yuhong Guo, Jieping Ye. "Object Detection in 20 Years: A Survey", Proceedings of the IEEE, 2023

Publication

<1 %

18

Zhanlin Ji, Yun Wu, Xinyi Zeng, Yongli An, Li Zhao, Zhiwu Wang, Ivan Ganchev. "Lung Nodule Detection in Medical Images Based on Improved YOLOv5s", IEEE Access, 2023

Publication

<1 %

19

Christopher D Duncan. "Advanced Crime Scene Photography", CRC Press, 2019

Publication

<1 %

20

Bin Hu, Nuoya Zhou, Qiang Zhou, Xinggang Wang, Wenyu Liu. "DiffNet: A Learning to Compare Deep Network for Product Recognition", IEEE Access, 2020

Publication

<1 %

21

Submitted to Technische Universität Dresden

Student Paper

<1 %

22

ijircce.com

Internet Source

<1 %

23	ijmpronline.com Internet Source	<1 %
24	"Advances in Aerial Sensing and Imaging", Wiley, 2024 Publication	<1 %
25	Submitted to Higher Education Commission Pakistan Student Paper	<1 %
26	"The Future of Artificial Intelligence and Robotics", Springer Science and Business Media LLC, 2024 Publication	<1 %
27	jusst.org Internet Source	<1 %
28	Submitted to Queen Mary and Westfield College Student Paper	<1 %

Exclude quotes On

Exclude matches

< 14 words

Exclude bibliography On