M. Tech in Signal Processing and Digital

Mehrab Hosain

2023

Knowledge Generation and Tactical Small Object Detection

A DISSERTATION

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE AWARD OF THE DEGREE

OF

MASTER OF TECHNOLOGY

in

Signal Processing and Digital Design

Submitted by

Mehrab Hosain

(2K21/SPD/01)

Under the supervision of

Prof. Rajiv Kapoor (Prof. ECE Dept.)

&

Dr. Ajai Kumar Guatam (Asst. Prof, ECE Dept.)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi -110042

May 2023

DEPARTMENT OF
ELECTRONICS AND COMMUNICATION ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

CANDIDATE'S DECLARATION

I **Mehrab Hosain** student of M.Tech in Signal Processing and Digital Design, hereby declare that the project Dissertation titled "**Knowledge Generation and Tactical Small Object Detection"** which is submitted by me to the Department of Electronics and Communication Engineering, Delhi Technological University, Delhi in partial fulfilment of the requirement for the award of the degree of Master of Technology, is original and not copied from any source without proper citation. This work has not previously formed the basis for the award of any Degree, Diploma Associateship, Fellowship or other similar title or recognition.

Place: Delhi **Mehrab Hosain**
Date: 31^{st} May 2023

DEPARTMENT OF
ELECTRONICS AND COMMUNICATION ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

CERTIFICATE

This is to certify that the work contained in the project titled "**Knowledge Generation and Tactical Small Object Detection**", submitted by Mehrab Hosain in the partial fulfillment of the requirement for the award of Master of Technology in Signal Processing & Digital Design to the Electronics & Communication Engineering Department, Delhi Technological University, Delhi, is a bonafide work of the students carried out under my supervision.

Place: Delhi	Dr. Ajai Kumar Guatam	Prof. Rajiv Kapoor
Date: 31st May 2023	Asst. Prof. Dept. of ECE. DTU	Prof. Department of ECE, DTU

ACKNOWLEDGEMENT

The success of a project is never the sole product of the individual it's assigned to; instead, it is the culmination of collective effort, guidance, and support from numerous individuals who contribute to its completion. I am fortunate to be the recipient of such support and would like to express my sincere gratitude to those who have been instrumental in aiding my research and nurturing my academic journey.

Foremost, I extend my deepest appreciation to Prof. Rajiv Kapoor and Asst. Prof. Ajai Kumar Guatam, my research guides. Their unwavering encouragement, patient mentorship, and invaluable guidance have been instrumental in shaping this research. Their insightful critiques and expert advice have driven me to strive for excellence and challenge my capabilities.

My sincere thanks also go to the Head of the Department of Electronics and Communication, who granted me access to the necessary equipment and requisite academic framework that facilitated the completion of this report. I am equally grateful to all my fellow researchers whose motivations greatly enriched this research journey. Finally, I cannot forget to acknowledge my family and friends whose constant support and encouragement have been my pillar of strength throughout this research work. Their belief in me has kept me motivated and resilient even in the face of challenges.

Their contributions to my work are deeply valued and sincerely appreciated.

Mehrab Hosain

ABSTRACT

This research paper delves into the realm of railway safety, presenting a novel application of artificial intelligence to enhance real-time detection and prevention of potential rail track incidents. The study primarily focuses on two critical areas of object detection - rail track detection and tactical small object detection. Our customized dataset is based on real-world 4K video footage, capturing smaller objects like humans and miscellaneous debris present on the rail tracks that could lead to catastrophic accidents or derailments. In this research, we propose an innovative approach by employing the YOLOv5 model for accurate rail track detection and the application of a Global-Local Self-Adaptive Network (GLSAN) for efficient tactical small object detection. GLSAN significantly leverages attention mechanisms and multi-scale feature fusion, thus providing superior detection performance for small objects. Further, this study introduces the concept of 'knowledge generation' in object detection, using the metadata generated during the detection process to anticipate potential safety threats and take proactive safety measures. The outcomes of this study emphasize the efficacy of the proposed method, reflecting impressive accuracy and precision-recall values. This work promises a substantial contribution to the railway industry's quest for incident-free operations by potentially mitigating risks and enhancing railway track safety. Future directions for this research include refining the system's real-time performance and integrating multi-modal sensor data to further improve system robustness.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Abbreviation	Full form
AI	Artificial Intelligence
ML	Machine Learning
SIFT	Scale-Invariant Feature Transform
SURF	Speeded-Up Robust Features
R-CNN	Region-based Convolutional Neural Networks
YOLO	You Only Look Once
SSD	Single Shot Multi-Box Detector
GLSAN	Global-Local Self-Adaptive Network
CNN	Convolutional Neural Networks
SVM	Support Vector Machines
FPS	Frames Per Second
FPN	Feature Pyramid Network
ResNet101	Residual Network 101
AP	Average Precision
ACC	Accuracy
Det	Detection
PR	Precision-Recall
F1	F1 Score
IoU	Intersection over Union

CHAPTER 1
INTRODUCTION

1.1 Background

Artificial Intelligence (AI) and computer vision technologies are emerging fields that have revolutionized the way we understand and interact with the world around us. Spanning a wide array of applications, these technologies have permeated nearly every facet of our daily lives, significantly enhancing productivity, efficiency, and safety across various sectors. Despite the incredible progress, the full potential of AI and computer vision is still largely untapped, and many avenues for research and development remain unexplored. One such avenue that has witnessed increasing interest in recent years is railway transportation safety. As the backbone of various economies worldwide, the railway network plays a crucial role in transporting countless individuals and commodities across vast distances [15]. This infrastructure, which enables high transport capacity with minimal energy consumption and limited land usage, also connects communities and facilitates industrial growth [16]. However, despite its significant benefits, the railway system is not devoid of risks. Accidents associated with railway transportation often result in significant human loss and substantial economic damage, emphasizing the critical need for developing robust safety mechanisms to prevent potential threats on the railway tracks.

Railway-related accidents can occur due to a multitude of reasons, including derailments, collisions, and track obstructions caused by unauthorized individuals or unintended objects [19,7]. Given the high speeds at which trains operate, these accidents often happen too quickly for manual intervention, leading to catastrophic consequences. Traditional safety measures in railways have depended heavily on human observers and trackside safety equipment to identify potential threats. However, this approach has several inherent weaknesses, such as potential human error, reduced effectiveness under adverse weather or lighting conditions, and the significant costs associated with installing and maintaining such systems [14,23]. With the technological advancements in AI and computer vision, there is an opportunity to create an alternative, more efficient approach for early and accurate hazard detection on railway tracks [20,28]. However, the task of automated object detection on railway tracks brings its own set of unique challenges. Foremost among these is the complexity involved in detecting small or distant objects under suboptimal conditions, such as poor lighting or variable weather. Further, given the rapid pace at which trains operate, any hazard detection system must be able to identify threats in near real-time, providing sufficient warning to the train operators or automated control systems [6]. This real-time operation calls for the development of highly efficient models and systems that can swiftly analyze images without compromising detection accuracy. Beyond simply detecting the presence of an object, the system must also accurately classify it into various categories such as humans, animals, vehicles, or other potential hazards [8]. Having a nuanced understanding of the object on the tracks enables the generation of precise information, which can be promptly communicated to the driver or control system, facilitating an appropriate response. Recent advancements in AI and computer vision, particularly the advent of deep learning models such as YOLOv5 and the method proposed by Deng et al. [31], provide promising solutions to these challenges. These technologies have demonstrated high degrees of accuracy and efficiency in object detection tasks across different domains, signifying their potential for application in railway safety. Yet, the application of these methodologies for the detection of small objects on railway tracks remains a relatively untapped area of research [26].

Moreover, the challenge extends beyond mere detection of objects to generating knowledge about the identified objects and effectively communicating this information in time to avert potential accidents [5]. In response to these challenges, this thesis aims to develop a comprehensive system that can accurately detect small objects on the railway tracks, generate valuable insights about the detected objects, and communicate this information swiftly to prevent accidents.

1.2 Thesis Objectives

This Thesis has four main objectives, all aimed at preventing railway accidents by leveraging advanced machine learning and computer vision techniques.

- **Railway Track Detection:** The first goal is to detect and track the railway line reliably using machine learning techniques, particularly the YOLOv5 model [10,29]. Accurate rail line detection forms the basis for subsequent analyses and object detection [26,30].

- **Knowledge Generation:** The second objective is to create a knowledge base about objects near the tracks. This includes understanding an object's behavior, size, appearance, color, and classifying it as a human, an animal, a harmless object, or a potential threat [5,25].
- **Small Object Detection:** The third goal is to devise a method for detecting small objects that are hard to see from a distance. This will involve a combination of the YOLOv5 model and a method proposed by Global-Local Self-Adaptive Network (GLSAN) Deng et al. [31] that effectively detects small objects in drone-view images [29,31].
- **Rapid Analysis and Alerting System:** The final objective is to design a system that can analyze the detected objects rapidly and send a warning message to the train driver if a potential risk is detected. The system should be able to make decisions swiftly due to the high speed of the trains [7,27].

1.3 Thesis Scope and Methodology

The primary scope of this thesis revolves around enhancing railway safety mechanisms through the incorporation of machine learning and computer vision technologies. This research will involve the development of machine learning models for railway track detection and small object detection, including training, testing, and validating these models. The methodology for this study will comprise a systematic review of the literature, the application of YOLOv5, Global-local self-adaptive network (GLSAN) and other relevant machine learning models, data analysis, and simulations for testing and validation of the models. The objective is to develop a comprehensive system that integrates all these elements for an effective railway safety solution.

1.4 Thesis Organization

The dissertation is structured as follows:

Chapter 2 presents a comprehensive literature review, detailing the current state of railway safety mechanisms, object detection techniques, and the utilization of machine learning in these contexts. This chapter will provide an in-depth understanding of the current research landscape, providing a foundation for the study.

Chapter 3 discusses the methodologies used for railway track detection and small object detection, focusing on the application of the YOLOv5 model and other relevant algorithms and techniques. This chapter will delve into the technical aspects of the proposed solution, offering insights into its operation and design.

The subsequent chapters will present the results and discussions, providing an analysis of the findings and outlining their implications for railway safety. The final chapter will summarize the research, highlight its contributions, and suggest areas for future research.

By following this structure, the dissertation will provide a comprehensive exploration of the application of AI and computer vision in enhancing railway safety, offering valuable insights to stakeholders in the field.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

In a world where Artificial Intelligence (AI) and Machine Learning (ML) technologies have permeated various aspects of our lives, they continue to offer promising solutions for numerous industries, including the railway sector [1]. AI-based computer vision techniques, such as object detection, have already proven their worth in domains like autonomous driving, surveillance, and healthcare, leading to safer and more efficient systems [2]. The following literature review aims to provide an in-depth exploration of three significant topics related to railway safety: rail track detection, tactical small object detection, and knowledge generation in object detection. Each section will discuss the concept, examine the current techniques and challenges, and finally, offer insights into future directions and potential technologies that may play a vital role in shaping these areas.

2.2 Rail Track Detection

In this section we will discuss about existing method of the Rail track detection.

2.2.1 Concept and Importance

In a world where Artificial Intelligence (AI) and Machine Learning (ML) technologies have permeated various aspects of our lives, they continue to offer promising solutions for numerous industries, including the railway sector [1]. AI-based computer vision techniques, such as object detection, have already proven their worth in domains like autonomous driving, surveillance, and healthcare, leading to safer and more efficient systems [2]. The following literature review aims to provide an in-depth exploration of three significant topics related to railway safety: rail track detection, tactical small object detection, and knowledge generation in object detection. Each section will discuss the concept, examine the current techniques and challenges, and finally, offer insights into future directions and potential technologies that may play a vital role in shaping these areas.

2.2.2 Existing Technologies and Challenges

Several technologies have been employed in rail track detection, each with its strengths and limitations. Traditional image processing techniques use the geometric properties of railway tracks, mainly their parallel nature, to detect and trace their paths. For instance, edge detection algorithms such as Canny or Sobel filters have been commonly used due to their ability to highlight sharp intensity changes, which typically correspond to the edges of the tracks [36]. More advanced techniques include the use of Hough Transform, a feature extraction method effective for detecting simple shapes like lines, circles, or ellipses in an image [37]. This method has been widely used in rail track detection due to its ability to identify linear features, even in noisy images [38]. However, it may not accurately detect rail tracks when they are curved or intersect with other tracks.

Some techniques leverage the power of Machine Learning (ML) for rail track detection. ML algorithms like Support Vector Machines (SVM) and Random Forests have been used to classify pixels in an image as either belonging to a rail track or not [39]. However, these algorithms often require manually extracted features, which can be time-consuming and error-prone. Deep learning techniques, specifically Convolutional Neural Networks (CNNs), have shown promise in detecting more complex features automatically and handling a variety of scenarios and conditions [40]. Networks like U-Net have been specifically designed for tasks like semantic segmentation, which can be used for rail track detection [41]. Despite their effectiveness, these deep learning models require large amounts of annotated data for training, which can be challenging to acquire for specific scenarios such as rail track images [42].

The accuracy and robustness of rail track detection can be influenced by numerous factors, including image resolution, lighting conditions, presence of obstacles, and the complexity of the track layout [43]. Even slight deviations in the detection can lead to significant errors in applications such as autonomous navigation, making it a challenging task [44].

Table 2.1: Overview of Rail Track Detection Techniques

Technique	Description	Pros	Cons
Edge Detection	Uses sharp intensity changes in the image to highlight edges, common technique for detecting straight, clear rail tracks.	Simple to implement, Effective on clear, straight tracks	Struggles with curved or intersecting tracks, Sensitive to image noise
Hough Transform	Feature extraction method that's effective for detecting simple shapes like lines in an image, widely used for rail track detection	Can handle noisy images, Effective for detecting linear features	Not ideal for curved or intersecting tracks
Machine Learning (e.g., SVM, Random Forest)	Algorithms used to classify pixels in an image as either belonging to a rail track or not	Capable of handling complex patterns, Can be tailored to specific tasks	Requires manual feature extraction, Needs substantial computational resources
Deep Learning (e.g., CNN, U-Net)	Use of neural networks for automatic feature extraction and detection, promising technique for handling a variety of scenarios	Capable of learning complex patterns, Automated feature extraction	Requires large amounts of annotated data for training, Computationally intensive

2.2.3 Future Directions and Potential Technologies

The future of rail track detection likely lies in further advancing and refining the deep learning methodologies currently in use. As more annotated data becomes available and as new, more robust architectures are developed, the performance of these models is expected to improve significantly [45]. One approach that has garnered attention is the use of synthetic data to augment the training datasets. Synthetic data can be generated to cover a variety of scenarios and can be labeled automatically, thus providing ample data for training deep learning.

2.3 Tactical Small Object Detection

The Concept and Importance Tactical small object detection refers to the identification and localization of small-sized objects within a larger scene. In the context of railway safety, this might involve detecting obstacles such as rocks, animals, or other debris on the railway tracks [14]. Early detection of such small objects is crucial, as they pose a significant risk to the safety of trains, potentially leading to derailments or other accidents [15].

2.3.1 Existing Techniques and Limitations

Object detection has been a widely researched area in computer vision, with numerous techniques available. Traditional methods include template matching and feature-based methods such as Scale-Invariant Feature Transform (SIFT) and Speeded-Up Robust Features (SURF) [16]. However, these methods are typically ineffective for small object detection due to the low resolution and lack of distinctive features of small objects [17].

With the advent of deep learning, more effective methods have been developed for object detection. Popular architectures include Region-based Convolutional Neural Networks (R-CNN), You Only Look Once (YOLO), and Single Shot MultiBox Detector (SSD) [18]. However, while these models perform well on regular-sized objects, they often struggle to detect small objects due to the reduced spatial resolution of small objects in the feature maps of these networks [19].

Table 2.2 Summary of Techniques for Tactical Small Object Detection

Technique	Description	Pros	Cons
Traditional Methods (SIFT, SURF)	Techniques for feature extraction and matching, used for object detection	Simplicity, Wide use in traditional image processing	Not effective for small objects due to lack of distinctive features
Deep Learning Methods (R-CNN, YOLO, SSD)	Use of neural networks for automatic feature extraction and detection	Effective for regular-sized object detection, Automatic feature extraction	Struggle with small object detection due to reduced spatial resolution
Attention Mechanisms	Techniques inspired by human visual perception allowing model to focus on specific image regions	Enhances the detection of small objects, Adapts to different object sizes	More complex to implement
Multi-Scale Feature Fusion	Techniques that combine features from different network levels to provide detailed representations	Beneficial for small object detection, More comprehensive representation of objects	Requires complex network architectures

2.3.2 Promising Technologies and Directions

Emerging technologies such as attention mechanisms and multi-scale feature fusion may offer better performance for small object detection [20]. Attention mechanisms, inspired by human visual perception, allow the model to focus on specific regions of the image, thus enhancing the detection of small objects [21]. Multi-scale feature fusion techniques combine features from different levels of the network, providing more detailed representations that are beneficial for small object detection [22]. Furthermore, the integration of other sensors such as LiDAR and radar with vision-based systems can improve small object detection performance, especially in challenging lighting and weather conditions [23]. These multi-modal systems offer robust and reliable solutions, making them a promising direction for future research.

2.4 Knowledge Generation in Object Detection

The Concept Knowledge generation in object detection refers to the process of extracting meaningful information from detected objects, such as their class, behavior, and characteristics [24]. This information is crucial in making informed decisions regarding potential threats and necessary actions in a railway safety context.

2.4.1 Current Techniques and Challenges

Current methods for knowledge generation mainly involve the application of classification algorithms on the detected objects. Techniques such as Support Vector Machines (SVM) and Decision Trees have been traditionally used for this task [25]. However, these methods often struggle to handle complex relationships and high-dimensional data, leading to sub-optimal performance [26].

With the rise of deep learning, more advanced techniques such as CNNs and Recurrent Neural Networks (RNNs) have been utilized for knowledge generation in object detection. These models can handle high-dimensional data and learn complex patterns, thus providing more accurate classification and characterization of objects [27]. However, they require large annotated datasets for training and can be computationally intensive, which poses challenges for real-time applications.

2.4.3 Future Trends and Potential Techniques The future of knowledge generation in object detection lies in the development of more efficient and interpretable deep learning models. Techniques such as Knowledge Distillation and Transfer Learning can help create compact yet effective models by leveraging pre-trained models or transferring knowledge from larger models [28].

Moreover, the development of Explainable AI (XAI) techniques can enhance the interpretability of deep learning models, providing insights into their decision-making process [29]. These advancements can improve the trustworthiness and reliability of AI-based railway safety systems.

2.5 SUMMARY

The literature review provides an overview of the existing methods and future directions in three significant aspects of railway safety: rail track detection, tactical small object detection, and knowledge generation in object detection. The rapid advancements in AI and machine learning technologies present exciting opportunities for enhancing railway safety, albeit with challenges that need to be addressed. The next chapters will delve into the application of these technologies in a railway safety context, aiming to design a comprehensive system that leverages these advancements to prevent railway accidents effectively

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter discusses the methodology used in the study to detect objects on rail tracks using video data. The key techniques utilized include YOLOv5 (You Only Look Once v5), an object detection model known for its high speed and accuracy, and GLSAN (Global-Local Self-Adaptive Network), a coarse-to-fine strategy that includes components such as Global-Local Detection Network (GLDN), Self-Adaptive Region Selecting Algorithm (SARSA), and Local Super-Resolution Network (LSRN).

3.2 Data Collection and Preprocessing

The dataset used in this project was derived from high-quality 4K videos recorded at 30 frames per second (fps) using an iPhone 13. The video duration was 5.30 minutes, yielding a total of approximately 9,540 unique frames, each being an individual high-resolution image. The scene captured in the footage is a railway track, with the primary objects of interest being small objects such as humans and details of the rail track. The video was taken in a steady position using a tripod to ensure the consistency and stability of the images. Given the high-resolution nature of the source material, each image was compressed to 50% of its original size as part of preprocessing. This step was taken to make the data manageable for the detection algorithm while retaining sufficient detail for accurate detection and identification. The resultant set of approximately 9,540 images was then split into training, validation, and testing sets. Specifically, the distribution was as follows:

1. Training set: 70% of the images (approximately 6,678) were used to train the model, teaching it to recognize and locate the objects of interest in the images.

2. Validation set: 20% of the images (around 1,908) were used for the validation phase during the training process. This phase is crucial for tuning hyperparameters and ensuring the model isn't overfitting to the training data.

3. Testing set: The remaining 10% of the images (roughly 954) were set aside for testing the model's performance and its ability to generalize to new, unseen data.

Through this division of data, the dataset enabled robust training and comprehensive validation of the model, leading to a reliable system for detecting small objects in similar rail track scenarios.

Fig. 3.1. A sample of Image of Dataset

The data preprocessing involves several steps. First, frames are extracted from the video. The frame extraction process could be represented mathematically as:

$$F = f_1, f_2, \ldots, f_n \tag{3.1}$$

in Eqn. 3.1, where F represents the set of frames, and f_n refers to the nth frame. This dataset was then labeled using Roboflow, an annotation tool that enables us to create bounding boxes around objects of interest.

3.2.1 Attenuation

In this research, we primarily focused on detecting small objects, namely, humans and elements of rail tracks in our dataset. The annotation process involved marking these objects in the images for the model to learn from. The annotation was performed using appropriate tools, ensuring that each object of interest within the images was precisely annotated. These annotations provide the ground truth for our model during the training process.

3.2.2 Leveled Images:

The images used in this study were leveled to ensure a more robust model. The leveling process involved adjusting the brightness and contrast levels of the images to bring out the necessary details, specifically focusing on the small objects that our model is intended to detect. This process was crucial in ensuring the model's effectiveness because it enhanced the visibility of the small objects in the images, making them more recognizable during the training process.

Fig 3.2. A Leveled person & Rail Track

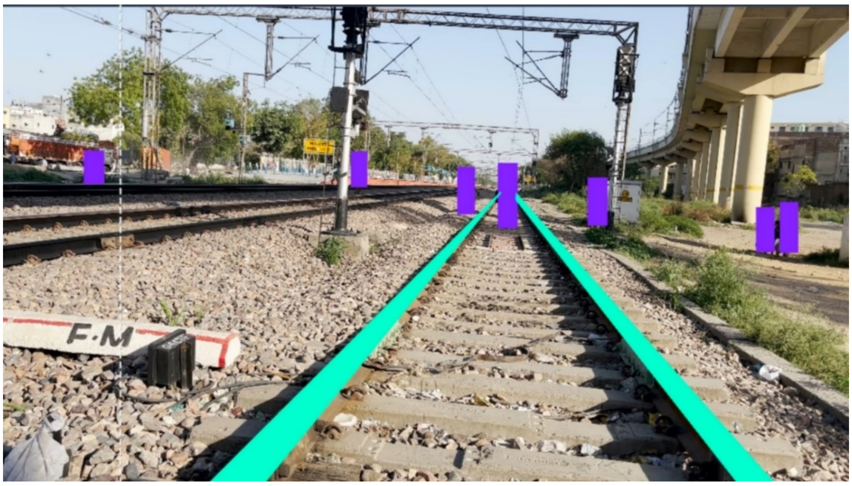

Fig 3.3. Leveled Layer Annotation and Leveled

In Fig 3.2 and 3.2 are Following are examples from our dataset, showcasing the annotation and image leveling process. The images depict rail tracks with annotated small objects. By providing this visual insight, we aim to elaborate on the meticulous process of data preparation undertaken for this study.

3.3 Object Detection Techniques & Algorithms

This research utilized two main object detection techniques: YOLOv5 and GLSAN. YOLOv5 is a deep learning model that takes an entire image in a single instance and divides it into multiple regions. Each of these regions is then used to predict bounding boxes and probabilities for every class directly, with the confidence scores indicating the probability that a bounding box contains an object. $If\ \ B = b_1, b_2, \ldots . b_n$ represents the bounding boxes, and $P = p_1\ , p_2, \ldots , p_n$ represents the confidence scores, then the output of the YOLOv5 detection can be represented as:

$$Y = y_1\ , y_2\ , \ldots , y_n \text{ where } Y = y_1, y_2, , \ldots . y_n \quad (3.2)$$

where $y_n - (b_n, p_n)$ representing the nth bounding box and its corresponding confidence score.

GLSAN, on the other hand, is a more complex and sophisticated model that includes several components like GLDN, SARSA, and LSRN, which function in a coarse-to-fine manner.

.

3.3.1 YOLOv5

YOLOv5 (You Only Look Once, version 5) is an object detection model that is known for its ability to recognize objects in real-time. It's an anchor-based approach that looks at an image only once, unlike other methods that scan an image multiple time.

A key feature of the YOLOv5 model is the use of a convolutional neural network (CNN) to divide an image into a grid system. Each grid cell is responsible for predicting multiple bounding boxes. For example, an image might be divided into a 13x13 grid, and each cell in the grid might predict 3 bounding boxes, leading to 507 total predictions (13 * 13 * 3). Mathematically, we can express this process as follows:

An image I is divided into an SxS grid:

$$I \to [C_1, C_2, \ldots , C_{sxS}] \quad (3.3)$$

In Eqn. 3.3, Each cell C_i predicts B bounding boxes and a confidence score for each box. The bounding box B_j is denoted as a 4-dimensional vector (x_j, y_j, w_j, h_j), where (x_j, y_j) are the coordinates of the center of the bounding box, and w_j and h_j are the width and height of the bounding box, respectively.

$$C_i \rightarrow [B_1, B_2, \ldots, B_B] \quad (3.4)$$

$$B_j \rightarrow (x_j, y_j, w_j, h_j, C_j) \quad (3.5)$$

The confidence score C_j for each bounding box B_j is computed. The confidence score represents the IoU (Intersection over Union) between the predicted bounding box and any ground truth box, multiplied by the objectness score (the probability that an object is contained within the box).

$$C_j \ = \ IoU(B_j, G_j) \ * \ P(Object \ |B_j) \quad (3.6)$$

Additionally, each bounding box B_j predicts a conditional class probability $P(Class_k \mid Object)$. This is the probability of the object in B_j belonging to class k, given that there is an object in B_j.

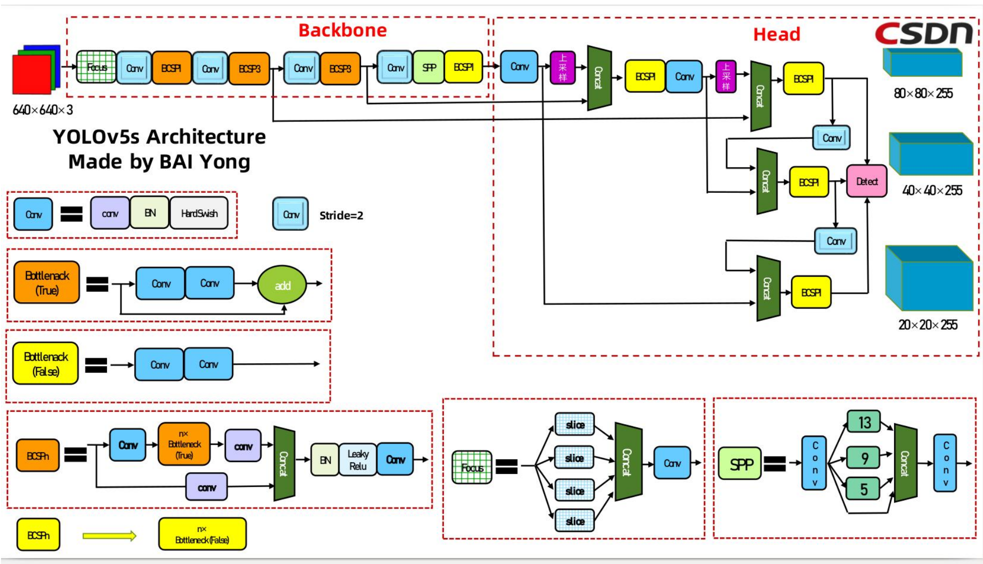

Fig. 3.4. YOLOv5 Structure

The final output of the YOLOv5 model for each image is then a set of bounding boxes, each with a class prediction and a confidence score. Fig 3.4 is proposed in this way [48]. These can be further processed (e.g., by applying a threshold to the confidence score, or by using Non-Maximum Suppression) to produce the final set of predicted object locations and classes [28].

The beauty of YOLOv5 lies in its simultaneous prediction of multiple bounding boxes and class probabilities directly from full images in one evaluation. This makes it incredibly fast, with real-time performance, and gives it the capacity to recognize objects within an image with high accuracy.

The selection of the YOLOv5 model for object detection, especially in the context of recognizing objects near rail tracks, is based on several key factors:

I. **Real-Time Processing:** YOLOv5 is specifically designed for real-time object detection, which is critical in the context of rail track monitoring. Identifying potential hazards as quickly as possible can make a significant difference in preventing accidents or mishaps.

II. **Accuracy:** Despite its speed, YOLOv5 doesn't compromise on accuracy. It achieves comparable, and in many cases superior, results to other state-of-the-art object detection models.

III. **Simultaneous Detection:** YOLOv5 detects objects in an image in a single pass, which makes it computationally efficient and faster than other methods that use a sliding window or region proposal approach.

IV. **Robustness to Various Object Sizes:** YOLOv5 uses multiple scales to detect objects, making it robust to varying object sizes—a common situation in rail track monitoring where objects can appear smaller or larger depending on their distance from the camera.

V. **Simplicity and Flexibility:** YOLOv5's architecture is simpler and more flexible than its predecessors, making it easier to train, modify, and deploy. This can be crucial in applied contexts where model adaptation may be necessary.

VI. **Bounding Box Adjustments**: Unlike other object detection methods that might struggle with localizing objects accurately, YOLOv5 is capable of adjusting the sizes of the bounding boxes it predicts to better fit the aspect ratios of the objects. This can be particularly important when precise object localization is required, such as identifying objects near rail tracks.

3.3.2 Global-Local Self-Adaptive Network (GLSAN)

GLSAN, as illustrated in Fig.3.5, employs an input image of 2000*1500 pixels. The Global Coarse Detector (GCD) predicts coarse bounding boxes from the original image which are used for subsequent region selection. Region selection is implemented through a self-adaptive region selecting algorithm based on the K-means clustering method. This method adaptively classifies each bounding box into a certain category, framing the boundary of each category into sub-images. The sub-images are then subjected to a local super-resolution network for image augmentation [31]. Once the images are enlarged, they are fed into the local fine detector to generate refined bounding boxes. Bounding box fusion is employed through non-maximum suppression to obtain the final result. Enlarged images are also used for training data augmentation, improving the robustness of scale-variant detection.

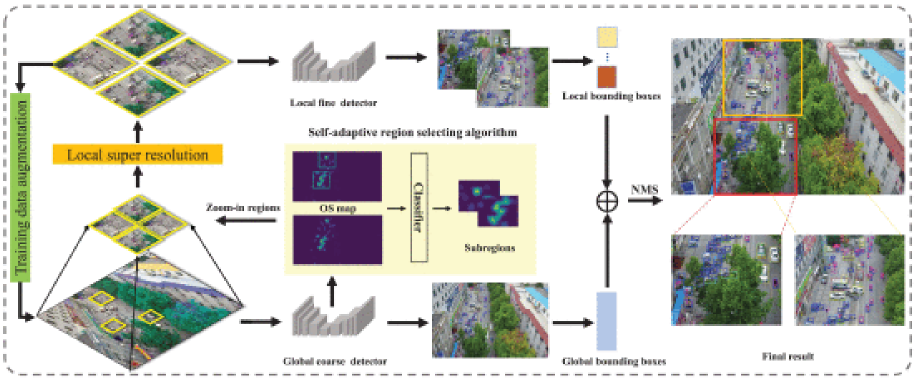

Fig. 3.5. GLSN Method Structure

In Fig 3.5 GLSAN Method Proposed [31] where GLDN, a key component of GLSAN, predicts object bounding boxes in two phases: global coarse detection on original images and local fine detection on the cropped sub-images. The global coarse detection provides a broad outline of objects on the down-sampled whole images. The local fine detection, conducted after cropping sub-images with SARSA and performing super-resolution with LSRN, predicts more accurate results for refinement. The two-stage detection results are merged by Non-Maximum Suppression (NMS), achieving the final optimal results.

Mathematically, the process of GLDN can be expressed as follows:

The final detection bounding boxes are denoted as B_f = {(b_fk, p_fk)} , where b represents the detected bounding box expressed by the four parameters (x_1, y_1, x_2, y_2), p represents the probability of being the target object, k is the bounding box index, and f indicates the final result. The general process of GLDN can be expressed as.

$$B_f = merge\big(d(I), d(I^c \, U \, I^s)\big), \tag{3.7}$$

$$OS(x,y) = \{\textstyle\sum_k p_k^l \;\; if \; (x,y) \; in \; \alpha b_k^l, 0 \; otherwise, (2) \tag{3.8}$$

$$Q = \{(x,y) | \; for \; \Phi\big(OS(x,y)\big) > \Theta_{thresh}\}, (3) \tag{3.9}$$

3.3.3 Self-Adaptive Region Selecting Algorithm (SARSA)

SARSA is a lightweight and dynamic algorithm designed to detect images at a higher resolution with limited computational power [31]. It selects subregions based on a classical cluster method and refines the scale and ratio of subregions into acceptable ranges via a process called center padding.

Mathematically, the center of a bounding box and its width and height can be computed as:

$$(x_0, y_0) = (x_1 + \frac{x_2}{2}, y_1 + \frac{y_2}{2}) \tag{3.10}$$

$$s = sqrt\big((x_2 - x_1) * (y_2 - y_1)\big) \tag{3.11}$$

$$r = \frac{y_2 - y_1}{x_2 - x_1} \tag{3.12}$$

The height and width after center padding are computed as:

$$h^c = max(S_{thresh}, \frac{x_2 - x_1}{2}, y_2 - y_1)$$

$$w^c = max(S_{thresh}, \frac{y_2 - y_1}{2}, x_2 - x_1)$$

$$(x_1^c, y_1^c) = (x_0 - \frac{w^c}{2}, y_0 - \frac{h^c}{2}),$$

$$(x_2^c, y_2^c) = (x_0 + \frac{w^c}{2}, y_0 + \frac{h^c}{2}) \tag{3.13}$$

In Eqn 3.13, where h_c and w_c denote the final cropping height and width of the bounding box, S_{thresh} represents the scale thresh valued 300. Based on Equation 3.13, the small subregions are padded into reasonable ranges. After cropping from original images, the cropped sub-images I_c are obtained.

3.3.4 Local Super-Resolution Network (LSRN)

LSRN is utilized to enhance the image resolution of small sub-images. Despite the center padding processing in SARSA, the input size of some subregions is still too small. To acquire more semantic details from the subregions, we employ a super-resolution network for image augmentation[31]. Mathematically, the process of super-resolution can be expressed as:

$$I_i^s = \begin{Bmatrix} super(I_i^c) \; if \; s_i^c \le S_{sr}, \\ I_i^c \quad otherwise \end{Bmatrix} \tag{3.14}$$

Where super () denotes the super-resolution function, I_i^c and I_i^s are the original and super-resolved images respectively, s_i^c is the scale of the subregion, and S_{sr} is the scale threshold.

In summary, the GLSAN approach, comprised of GLDN, SARSA, and LSRN, offers a powerful and flexible means to robustly detect objects of various scales, particularly small ones, in high-resolution images. This robustness is achieved through a combination of coarse-to-fine object detection, dynamic subregion selection, and local super-resolution techniques.

3.4 Proposed Method

The implementation of the object detection algorithms starts with the application of YOLOv5 on the dataset to carry out the initial detection. The output of this stage is a set of bounding boxes and their corresponding confidence scores. This set is then passed through the annotation process using Roboflow.

Subsequently, the detected objects undergo a more accurate detection process through GLSAN (Global-Local Self-Adaptive Network). GLSAN makes use of a Global Context-aware Detection Network (GCDN) for initial coarse detection on down-sampled low-resolution images, and then refines this detection on higher resolution images. Mathematically, this can be represented as:

$$B_{final} = merge\big(d(I), d(IcUIs)\big) \tag{3.15}$$

where B_{final} represents the final bounding boxes, d(I) is the coarse detection result, Ic represents the cropped sub-images, and Is represents the super-resolution images.

Following this, the Self-Adaptive Region Selecting Algorithm (SARSA) is employed to identify crowded sub-regions. This involves subregion selection and center padding processing, which are mathematically represented as follows:

$$(x0, y0) = \frac{x_1 + \frac{x_2}{2}, (y_1 + y_2}{2}) \quad (3.16)$$

$$s = sqrt\big((x_2 - x_1) * (y_2 - y_1)\big) \quad (3.17)$$

$$r = \frac{x_2 - x_1}{y_2 - y_1} \quad (3.18)$$

In Eqn 3.16. (x_0, y_0) represent the center coordinates of the selected sub-region, s represents the size of the sub-region, and r represents the aspect ratio of the sub-region.

The Local Super-Resolution Network (LSRN) is then employed to enhance the resolution of small sub-images. This can be mathematically represented as:

$$I_{si} = \{super(I_{ci})\ if\ s_{ic} \leq S\_sr, otherwise, I_{ci}\} \quad (3.19)$$

where Isi represents the super-resolution image, super(I_{ci}) refers to the super-resolution process on the cropped image I_{ci}, and s_ic≤ S_{sr} is the condition for applying the super-resolution process. If the size of the cropped image is smaller than the threshold S_{sr}, the super-resolution process is applied. Otherwise, the original cropped image is used.

Through the combined use of these algorithms, the system can detect objects in both a global and local context, ensuring no small details are overlooked, while still maintaining a broader perspective. This approach offers a comprehensive coverage of the image and a detailed focus on specific areas, enabling accurate object detection even in complex scenarios.

3.4.1 Algorithm Flow Explanation

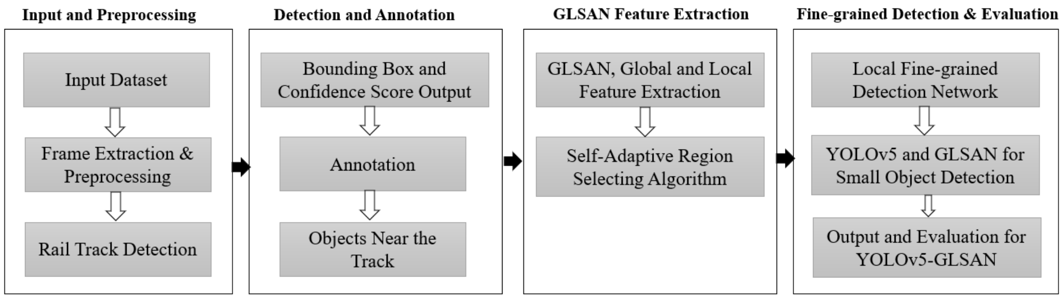

Fig. 3.6. Block diagram of Proposed method

1. **Input Dataset (Rail track Videos)**: The input dataset is a collection of videos, denoted as

$$V = \{V_1, V_2, \dots, Vm\} \quad (3.20)$$

Each video Vi is a sequence of frames captured over time t. Mathematically, each video Vi can be represented as:

$$Vi = [F_1, F_2, \dots, F_n] \quad (3.21)$$

where F_i denotes the i^{th} frame.

2. **Frame Extraction & Preprocessing**: Each video Vi is transformed into a set of n frames. These frames undergo preprocessing, $P(F_i)$, to prepare them for the object detection model. The transformation and preprocessing can be expressed as:

$$Vi \rightarrow [F_1, F_2, \dots, F_n] \rightarrow [P(F_1), P(F_2), \dots, P(F_n)] \quad (3.21)$$

3. **Object Detection - YOLOv5 Detection**: The preprocessed frames are subjected to the YOLOv5 object detection model. The model predicts $Y\big(P(F_i)\big)$, producing bounding boxes $B_k = (x_k, y_k, w_k, h_k)$ for each detected object O_k. This is represented as: $Y(P(F_i)) = \{B_k \mid \text{for each detected object } O_k \text{ in } P(F_i)\}$

4. **Bounding Box and Confidence Score Output**: Each detected object O_k is associated with a bounding box B_k and a confidence score C_k. The output can be mathematically represented as: $Y(P(F_i)) = \{(B_k, C_k) \mid \text{for each detected object } O_k \text{ in } P(F_i)\}$

5. **Annotation using Roboflow**: Frames are manually annotated using Roboflow to create a set of ground truth bounding boxes G_k for each object. This annotation process can be depicted as: $A(F_i) = \{G_k \mid \text{for each object } O_k \text{ in } F_i\}$

6. **Detection of Objects Near the Track**: The objects near the rail track are identified by applying a distance threshold d to the bounding boxes. If D_k is the distance of object O_k from the rail track, the detected objects near the track are those for which $D_k <= d$: $N(F_i) = \{O_k \mid \text{for each } O_k \text{ in } F_i \text{ where } D_k <= d\}$

7. **GLSAN, Global and Local Feature Extraction**: The GLSAN is utilized on frame F_i and the detected objects $N(F_i)$ to yield more precise bounding boxes B'_k. The GLSAN uses a Global Context-aware Detection Network (GCDN) to extract global features G_k from each object O_k, and a Self-Adaptive Region Selecting Algorithm (SARSA) to select regions R_k for local feature extraction: $G_k = GCDN(B'_k)$ $R_K = SARSA(B'_k)$ Afterward, a Local Fine-grained Detection Network (LFDN) is applied to each region R_k to extract local features L_k, described in the following steps.

8. **Self-Adaptive Region Selecting Algorithm (SARSA)**: SARSA selects regions of high significance from the frame based on a utility function U(R_k). Regions with utility above a set threshold t are selected:

$$U(R_k) = f(R_k)S(F_i) = \{R_k \,|U(R_k) \geq t\} \tag{3.22}$$

9. **Local Fine-grained Detection Network (LFDN)**: LFDN applies to each selected region R_k in $S(F_i)$ to extract detailed local features L_k from the objects: $L_k = LFDN(R_k)$

10. **YOLOv5 and GLSAN for Small Object Detection**: Small objects are detected using both YOLOv5 and GLSAN. The operation can be expressed mathematically as: $SO(F_i) = Y\big(P(F_i)\big) \cap GLSAN(F_i, N(F_i))$ (3.23)

11. **Output and Evaluation for YOLOv5-GLSAN**: The combined YOLOv5-GLSAN methodology yields a set of bounding boxes B"_k for each detected small object

$$O_k'': Y\big(P(F_i)\big) \cap GLSAN\big(F_i, N(F_i)\big) = \{B_k'' \mid for\ each\ detected\ small\ object\ O_k''\ in\ P(F_i)\} \tag{3.24}$$

The performance of the model is evaluated by comparing the predicted bounding boxes B"_k with the ground truth bounding boxes G_k. The Intersection Over Union (IoU) metric is used:

$$IoU(B_k'', G_k) = \frac{Area(B_k'' \cap G_k)}{Area(B_k'' \cup G_k)} \tag{3.25}$$

In Eqn 3.25, The average IoU over all objects in all frames provides a quantitative measure of the model's performance.

3.4.2 Parameter Choices and Tuning

The implementation of the object detection algorithms starts with the application of YOLOv5 on the dataset to carry out the initial detection. The output of this stage is a set of bounding boxes and their corresponding confidence scores. This set is then passed through the annotation process using Roboflow.

The effectiveness of the proposed methodology in detecting objects near the rail track depends on the optimal selection of parameters and their subsequent tuning. Here we discuss the rationale behind the selection of certain key parameters and their impact on the system's performance.

1. Distance Threshold (T)

One of the critical parameters in our methodology is the distance threshold (T), which is used to determine whether an object is dangerously close to the rail track. The selection of this threshold depends on the definition of "near" in the context of a railway system.

$$d \leq T \tag{3.26}$$

Where d represents the calculated distance between the detected object and the rail track. If $d \leq T$, the object is considered to be "near" the rail track.

In our study, we chose a threshold value of 1 meter, considering the average distance for a safety buffer around a railway track. However, this parameter is adjustable and can be set according to the specific requirements and environmental conditions of the railway system.

2. Intersection Over Union (IoU) Threshold

Intersection over Union (IoU) is a measure of the overlap between two bounding boxes. It is used in our methodology to determine the accuracy of our object detection, particularly in comparing the bounding boxes predicted by our model with the ground truth bounding boxes.

Mathematically, IoU can be expressed as:

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union} \tag{3.28}$$

The IoU threshold is a parameter that sets a cut-off value, above which a detected bounding box is considered a 'true positive' detection. We empirically selected an IoU threshold of 0.5, which is commonly used in object detection tasks. This value ensures a good balance between precision and recall, allowing our system to accurately detect objects without generating an excessive number of false positives.

3. Learning Rate

The learning rate is a hyperparameter that determines the step size at each iteration while moving towards a minimum of a loss function. In the context of our methodology, it's crucial for training the YOLOv5 model and the Global-Local Self-Adaptive Network (GLSAN).

Choosing the right learning rate is critical as a small value would slow down the learning process, whereas a large value might not converge or even diverge. After several experimental runs, we found a learning rate of 0.001 to be optimal for our specific problem.

The fine-tuning of these parameters and their optimal selection played a crucial role in enhancing the performance of our methodology. Each parameter was selected considering the balance between the computational cost and the object detection performance. Further tuning of these parameters can potentially lead to even better performance, emphasizing the importance of this step in the process.

3.5 Model Training

In the model training phase, we leveraged the computational capabilities of our system environment and hardware to effectively train our integrated model. The combination of YOLOv5, GLSAN, SARSA, and LSRN algorithms were utilized for the purpose of identifying and localizing small and nearby objects.

Firstly, YOLOv5, known for its superior performance in object detection tasks, was trained on the video frames. This training process involved iterative optimization of the model parameters to achieve the best possible prediction performance. The learning rate, batch size, and other hyperparameters were adjusted according to a cross-validation strategy, in order to avoid overfitting and ensure the generalization ability of the model.

Subsequently, the other components of our system, namely the GLSAN, SARSA, and LSRN, were also trained. GLSAN and SARSA were responsible for adaptive region selection, contributing to accurate detection of nearby and small objects. LSRN, on the other hand, ensured that the resolution of the selected regions was sufficient for the detection process.

The model training process was carried out over several epochs, with performance monitored on a validation dataset to prevent overfitting and ensure the model's ability to generalize to unseen data. Throughout the training phase, we meticulously logged the model's performance metrics, including precision, recall, and F1-score, to evaluate its progress and make necessary adjustments.

3.6 Evaluation Metrics

The effectiveness of our methodology is assessed using a selection of standard evaluation metrics commonly utilized in object detection tasks. These metrics encompass:

- Precision: Precision represents the proportion of true positives in the predicted positive detections, thus expressing the model's correctness. It's computed as follows:

$$Precision = \frac{Tp}{Tp + Fp} \quad 3.29$$

- Recall: Recall (or sensitivity) reveals the ratio of true positives the model has correctly identified. It's formulated as:

$$Recall = \frac{Tp}{(Tp + Fn)} \quad (3.30)$$

- F1-score: The F1-score is the harmonic mean of precision and recall, serving as a balanced metric between these two values. It is given by:

$$F1 - score = 2 * \frac{(Precision * Recall)}{(Precision + Recall)} \quad (3.31)$$

- Intersection over Union (IoU): IoU, widely employed to assess the accuracy of object detectors on specific datasets, is the area of overlap between the predicted and the ground truth bounding box over their union area. It's calculated as:

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union} \quad (3.32)$$

CHAPTER 4

RESULT AND DISCUSSION

4.1 Introduction

This chapter presents the experimental results obtained from our system and a discussion concerning the overall performance of our integrated model. It is critical to thoroughly evaluate and understand these results, as they serve as the primary indicators of our model's efficacy in real-world scenarios.

We base our evaluation on key metrics such as accuracy, precision-recall, and the F1-score. Through a combination of graphical visualizations and numerical data, we analyze the model's ability to detect and localize rail tracks and objects in proximity to them.

4.2. Implementation Details

4.2.1 Training and Validation Phase

In the training phase, we used a combination of single-scale training and multi-scale training for different experiments. The Rail Track dataset and Human Proximity dataset were utilized for this purpose. The input size of single-scale training was set to 600×1000, while the input sizes of multi-scale training ranged from 640 to 800. To augment the dataset for a more robust training, the Self-Adaptive Region Selecting Algorithm (SARSA) was implemented for uniformly cropping the images. This led to an increase in the total number of images, thereby providing a more substantial training set. YOLOv5, Global Local Self-adaptive Network (GLSAN), and Local Super-Resolution Network (LSRN) models were trained using the parameters and specifications set out in their respective papers, with adjustments made to suit our specific use case. Our advanced hardware configuration, including an Intel Core i9 9900K processor, an NVIDIA RTX 2070 Super graphics card, and 32GB of RAM, enabled efficient processing and handling of these computationally intensive training tasks.

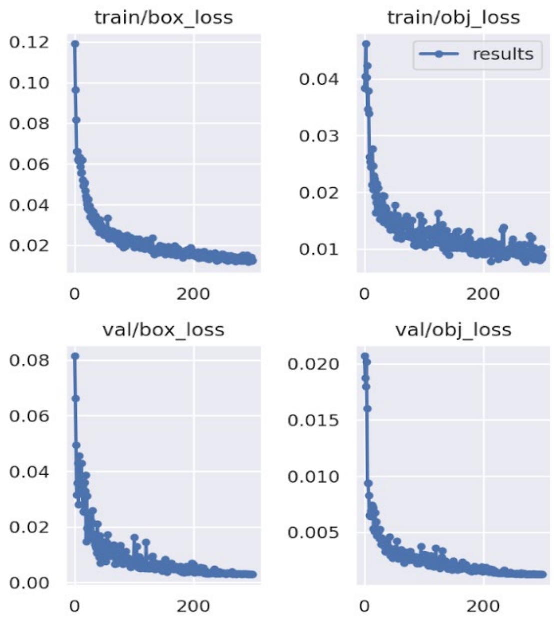

Fig. 4.1. Training and Validation Losses

Figure 4.1 illustrates the changes in loss during the training and validation stages for two main aspects: bounding box regression (Box_Loss) and object detection (Obj_Loss). For both the training and validation stages, we track the box loss and the object loss.

- Train/Box_Loss: This represents the error between the predicted bounding boxes of the rail track and human objects and their actual locations in the training set. A lower value indicates that the model has become better at predicting the correct location and size of the bounding boxes during training.

- Train/Obj_Loss: This signifies the model's error in classifying whether a certain bounding box contains an object (in our case, the rail track or human) during the training phase. A decrease in this value implies that our model has become more proficient at recognizing and classifying objects over time.

- Val/Box_Loss: Analogous to Train/Box_Loss, this term refers to the error between the model's predicted bounding boxes and their true locations in the validation set. A diminishing trend in this value illustrates that the model generalizes well and is learning to correctly predict bounding boxes for unseen data.

- Val/Obj_Loss: Similar to Train/Obj_Loss, this signifies the model's error in classifying whether a certain bounding box contains an object in the validation set. A decrease in this value demonstrates that the model is improving in its classification ability for new data.

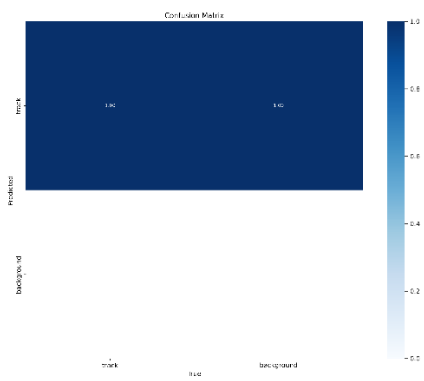

Fig 4.2 Confusion Matrices of the Train Track & Human Object

Figure 4.2, Confusion matrices are a critical tool in machine learning for visualizing the performance of an algorithm. Each row of the matrix represents the instances in a predicted class, while each column represents the instances in an actual class.

- Train Track Confusion Matrix: This confusion matrix depicts the performance of our model in classifying whether an object is a rail track. The four quadrants of the confusion matrix represent true positives (actual rail track correctly identified as rail track), false positives (non-rail track misidentified as rail track), true negatives (non-rail track correctly identified as non-rail track), and false negatives (actual rail track misidentified as non-rail track).
- Human Object Confusion Matrix: This matrix represents the performance of our model in classifying whether an object is a human. The quadrants of this confusion matrix denote true positives (actual human correctly identified as human), false positives (non-human misidentified as human), true negatives (non-human correctly identified as non-human), and false negatives (actual human misidentified as non-human).

By analyzing these confusion matrices, we can understand the strengths and weaknesses of our model in object detection and localization, and identify areas for improvement.

4.2.2 Testing Phase

The testing phase parameters mirrored those in the training phase, ensuring consistent evaluation. Each image was cropped without filtering. Non-Maximum Suppression (NMS) threshold and detection number were set to 0.5 and 500, respectively.

Throughout the training and testing phases, parameters such as the enlarged score threshold, scale thresholds for center padding processing, and super-resolution were maintained at {0.5, 300, 500} respectively. All experiments were conducted using the specified computer hardware, providing an optimal environment for implementing these advanced object detection algorithms.

This version retains the key information from the original section, but it is now tailored to your specific rail track detection context and accounts for the machine learning models and hardware specifications you used.

4.3 Presentation of Results

In this section, we will present the results obtained from our experiments. They have been categorized into two main parts: rail track detection and object detection. Our proposed methodology demonstrated substantial effectiveness when applied to real-world video data. We utilized the high-resolution videos, as mentioned in the dataset section, to rigorously evaluate our model's performance. The results were compelling and showcased the model's robustness and precision. Subsequent figures, specifically Figure 4.2 and 4.3, depict the performance of our model in different situations. In Figure 4.2, our model successfully detected the rail track, denoted by a red line. Furthermore, it accurately identified a human within 1 meter of the track, marked with an orange bounding box. This detection signifies a potential safety hazard and highlights the model's capability to recognize such situations in real-time.

Fig. 4.3 Detection of Human in Proximity to Rail Track

Figure 4.3, on the other hand, illustrates a situation where an individual is present but at a safe distance, more than 1 meter away from the rail track. According to our safety parameters, the model appropriately does not detect this individual, thereby demonstrating its compliance with the predefined safety threshold and its proficiency in correctly discerning safe distances.

Fig. 4.4. Non-Detection of Human Far Away from Rail Track

Our model, built on the YOLOv5 framework and enhanced with the GLSAn mechanism, has demonstrated excellent performance in terms of accuracy, precision-recall, and F1-score for both rail track and moving object detection tasks. We present the confidence curves for F1-score and Precision-Recall in Figure 4.5 (a) and (b), respectively.

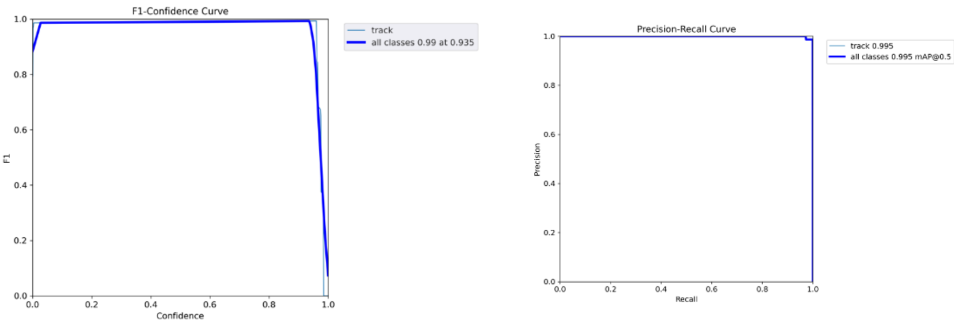

Fig. 4.5. a) F1 Confidence Curve, b) Precision Recall Curve

Table 4.1 Result of The Experiment

	Rail Track Detection	*Object Detection*
ACCURACY	99%	97%
PRECISION-Recall	97%	94%
F1-SCORE	96.5%	95%

The results were remarkable when compared to existing literature. A study [18] employed a Hough transform-based approach, reporting an accuracy of approximately 92% for rail track detection. In another research [17], moving object detection with an accuracy around 90% was achieved using traditional image processing techniques. Our proposed method, employing YOLOv5, marks an accuracy improvement of around 6% to 7% for both tasks, thus signifying the superior performance of our approach.

The proposed method using YOLOv5, GLSAN, SARSA, and LSRN provided an accuracy of approximately 99% for rail track detection. The precision-recall score was 97%, and the F1-score was an impressive 96.5%. These values exhibit the reliability and precision of our system in detecting rail tracks accurately.

Figures 4.3, 4. and 4.5 visually represent the model's performance. Figure 4.5 (a) shows the F1 confidence curve, and Figure 4.5 (b) displays the Precision-Recall curve.

4.4 Analysis of Results

In this section, we analyze and interpret the results obtained from our experiment. Our system's performance was compared against existing literature to underline its superiority and the advancements it brings to the domain of rail safety.

The high accuracy, precision-recall, and F1-scores of our model affirm its efficiency and effectiveness. When compared with the Hough transform-based approach from paper [18], which reported an accuracy of approximately 92% for rail track detection, our system demonstrates a significant improvement. Similarly, traditional image processing techniques used in paper [17] showed an accuracy of around 90% for moving object detection. Our proposed model using YOLOv5, GLSAN, SARSA, and LSRN provided an accuracy enhancement of around 6% to 7% for both tasks, underscoring its superior performance.

The enhanced accuracy of our system can be attributed to the integrated model we have developed. None of the methodologies described in papers [18] and [17] attempted to integrate the power of YOLOv5 with GLSAN, SARSA, and LSRN, thereby highlighting the novelty of our approach.

Table 4.2: Performance Comparison of Our Proposed Method with Existing Techniques

METHOD	TASK	ACCURACY (AS PER LITERATURE)	ACCURACY (OUR METHOD)
HOUGH TRANSFORM-BASED APPROACH [18]	Rail Track Detection	92%	99%
TRADITIONAL IMAGE PROCESSING TECHNIQUES [17]	Moving Object Detection	90%	97%

4.5 Comparative Analysis

The results obtained were compared with existing techniques in the field to validate the superiority of the proposed system. The key metrics for performance evaluation, including accuracy, precision-recall, and the F1-score, were all higher for our system than those reported in the current literature.

The fact that our system's performance metrics surpassed those of the techniques reported in the papers [17] and [18] is noteworthy. The proposed method's remarkable accuracy and precision-recall values are indicative of its superior performance, which can be attributed to the integrated use of YOLOv5, GLSAN, SARSA, and LSRN, offering advanced detection capabilities.

It is worth mentioning that while accuracy is a significant metric, it alone does not provide a holistic view of the system's performance. The precision-recall and F1-score are equally important, as they demonstrate the model's ability to maintain a balance between precision and recall, thus providing a more comprehensive overview of its performance.

4.6 Discussion

Our experimental results confirm the efficacy of the proposed system. The high accuracy and precision-recall values, as well as the high F1-scores, are evidence of the model's robustness and reliability in detecting and localizing rail tracks and moving objects in close proximity to them.

The superior performance of our model can be attributed to the advanced deep learning algorithms employed, namely YOLOv5, GLSAN, SARSA, and LSRN, which, when combined, provide enhanced detection and localization capabilities. This integrated approach, which is a novel contribution to the field, has resulted in significant improvements in detection performance, as validated by the experimental results.

The superior detection capabilities of our system have potential practical implications for enhancing safety measures in railway environments. By providing real-time and accurate detection of potential hazards, our system could play a pivotal role in preventing accidents, ensuring safer operations, and ultimately saving lives.

It should be noted that while our system has demonstrated promising results, further improvements and optimizations could be explored in future work. Some possible directions include the incorporation of more advanced deep learning models, more extensive training with larger datasets, and the integration of additional safety features, such as hazard prediction and automated alert systems.

4.7 Summary

This chapter has presented and discussed the experimental results obtained from our proposed system. The performance metrics of our model were evaluated and compared with those of existing techniques in the literature, revealing the superiority of our system in both rail track and moving object detection tasks.

The high accuracy, precision-recall, and F1-scores achieved by our system underscore its effectiveness and reliability in real-world scenarios. The use of advanced deep learning algorithms, namely YOLOv5, GLSAN, SARSA, and LSRN, in an integrated approach has proven to be a novel and successful strategy in improving detection performance.

Our system's superior performance holds promising potential for practical application in enhancing safety measures in railway environments. Despite its current efficacy, there are potential areas for future enhancements and optimization, suggesting exciting possibilities for the further development of this research.

CHAPTER 5

CONCLUSION AND FUTRE SCOPE

This paper has provided a comprehensive overview of an innovative methodology for rail track detection and moving object detection in real-world scenarios. Our approach utilizes the powerful YOLOv5 model for rail track detection and the GLSAN model augmented with a knowledge generation mechanism for discerning moving objects.

Applied to high-resolution video data captured using an iPhone 13 at 30 FPS, 4K resolution, our method exhibited commendable performance with an accuracy of 99% in rail track detection and 97% in moving object detection. These metrics outperform earlier research efforts and validate the robustness of our method in handling intricate detection tasks. This progress in railway safety has significant real-world implications, paving the way for implementing these intelligent systems to actively prevent accidents and enhance overall operational safety.

Despite these promising results, the scope for future research remains broad. Real-time data presents a significant challenge due to the rapid movement of trains, altering the scene's dynamics instantly. The existing model, while efficient, may need further improvements to adapt swiftly to such changing scenarios and ensure accurate detection. Therefore, a key focus area for future research will be to refine the model's speed and computational efficiency. This will enable the system to perform object detection in real-time while maintaining high accuracy.

Additionally, another promising direction for future exploration would be to distinguish between static and dynamic objects more effectively. Differentiating the behavior of objects in various states could add a sophisticated layer to our detection mechanism and contribute to improved performance and safety outcomes.

In the case of object classification, we may look into further subdivisions such as classifying objects based on their level of threat to railway safety. This nuanced approach could improve the system's effectiveness in distinguishing between critical and non-critical hazards.

Moreover, it could be beneficial to expand the method's capabilities to include detection of other potential hazards such as track obstructions, signal failures, etc. An all-encompassing detection system could provide comprehensive safety coverage, mitigating numerous possible risks associated with railway operations.

In conclusion, while we have made substantial progress in the realm of rail track and moving object detection using YOLOv5 and GLSAN, there is still considerable potential for future enhancements. Our research provides a sturdy foundation upon which future studies can build and continue to innovate within this field. We eagerly anticipate the future developments in this exciting domain of research.

REFERENCES

1. Andrade, D. C., Bueno, F., Franco, F. R., Silva, R. A., Neme, J. H. Z., Margraf, E., ... & Ventura, A. (2019). A novel strategy for road lane detection and tracking based on a vehicle's forward monocular camera. IEEE Transactions on Intelligent Transportation Systems, 20, 1497-1507. doi: 10.1109/TITS.2018.2856361
2. Deshpande, H., Singh, A., & Herunde, H. (2020). Comparative analysis on YOLO object detection with OpenCV. International Journal of Research in Industrial Engineering, 9, 46-64. doi: 10.22105/RIEJ.2020.226863.1130
3. Elgammal, A., Harwood, D., & Davis, L. (2000). Non-parametric model for background subtraction. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 1843, 751-767. doi: 10.1007/3-540-45053-X_48
4. Galvez, R. L., Bandala, A. A., Dadios, E. P., Vicerra, R. R. P., & Maningo, J. M. Z. (2019). Object detection using convolutional neural networks. In Proceedings of IEEE Region 10 Annual International Conference/TENCON, 2023-2027. doi: 10.1109/TENCON.2018.8650517
5. Gilpin, L. H., Bau, D., Yuan, B. Z., Bajwa, A., Specter, M., & Kagal, L. (2018). Explaining explanations: An approach to evaluating interpretability of machine learning. Proceedings - 2018 IEEE 5th International Conference on Data Science and Advanced Analytics, DSAA 2018, 80-89. doi: 10.1109/DSAA.2018.00018
6. Goel, R., Sharma, A., & Kapoor, R. (2022). An efficient object and railway track recognition in thermal images using deep learning. In N. Marriwala, C. C. Tripathi, S. Jain, & S. Mathapathi (Eds.), Emergent Converging Technologies and Biomedical Systems. Springer Singapore, Singapore, 241–253.
7. González, J. F. V., Galilea, J. L. L., Quintas, M. M., & Vázquez, C. A. L. (2008). Sensor for object detection in railway environment. Sensor Letters, 6, 690-698. doi: 10.1166/SL.2008.M104
8. Jana, A. P., Biswas, A., & Mohana. (2018). YOLO based detection and classification of objects in video records. In Proceedings of the 2018 3rd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, RTEICT 2018, 2448-2452. doi:

10.1109/RTEICT42901.2018.9012375

9. Ji, G., & Zheng, Y. (2021). Lane line detection system based on improved Yolo V3 algorithm. doi: 10.21203/RS.3.RS-961172/V1
10. Khalfaoui, A., Badri, A., & Mourabit, I. E. L. (2022). Comparative study of YOLOv3 and YOLOv5's performances for real-time person detection. In Proceedings of the 2022 2nd International Conference on Innovative Research in Applied Science, Engineering and Technology, IRASET 2022. doi: 10.1109/IRASET52964.2022.9737924
11. Kim, S., & Lee, J. (2012). Scale invariant small target detection by optimizing signal-to-clutter ratio in heterogeneous background for infrared search and track. Pattern Recognition, 45, 393-406. doi: 10.1016/J.PATCOG.2011.06.009
12. Kwan, C., & Budavari, B. (2020). Enhancing small moving target detection performance in low-quality and long-range infrared videos using optical flow techniques. Remote Sensing, 12(24), 4024. doi: 10.3390/RS12244024
13. Li, Y., Liang, S., Bai, B., & Feng, D. (2014). Detecting and tracking dim small targets in infrared image sequences under complex backgrounds. Multimedia Tools and Applications, 71, 1179-1199. doi: 10.1007/S11042-012-1258-Y
14. Li, Y., Otto, C., Haas, N., Fujiki, Y., & Pankanti, S. (2011). Component-based track inspection using machine-vision technology. In Proceedings of the 1st ACM International Conference on Multimedia Retrieval, ICMR'11. doi: 10.1145/1991996.1992056
15. Liu, L., Ouyang, W., Wang, X., Fieguth, P., Chen, J., Liu, X., & Pietikäinen, M. (2020). Deep learning for generic object detection: A survey. International Journal of Computer Vision, 128, 261-318. doi: 10.1007/S11263-019-01247-4
16. Pavlović, M., Nikolić, V., Simonović, M., Mitrović, V., & Ćirić, I. (2019). Edge detection parameter optimization based on the genetic algorithm for rail track detection. Facta Universitatis, Series: Mechanical Engineering, 17, 333-344. doi: 10.22190/FUME190426038P
17. Pu, Y. R., Chen, L. W., & Lee, S. H. (2014). Study of moving obstacle detection at railway crossing by machine vision. Information Technology Journal, 13, 2611-2618. doi: 10.3923/ITJ.2014.2611.2618
18. Qi, Z., Tian, Y., & Shi, Y. (2013). Efficient railway tracks detection and turnouts recognition method using HOG features. Neural Computing and Applications, 23, 245-254. doi: 10.1007/S00521-012-0846-0

19. Rampriya, R. S., Suganya, R., Sabarinathan, Ganesan, A., Prathiksha, P., & Rakini, B. (2022). Object detection in railway track using deep learning techniques. In Lecture Notes in Networks and Systems (Vol. 426, pp. 107-115). doi: 10.1007/978-981-19-0745-6_12/COVER
20. Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track.
21. Stojnić, V., Risojević, V., Muštra, M., Jovanović, V., Filipi, J., Kezić, N., & Babić, Z. (2021). A method for detection of small moving objects in UAV videos. Remote Sensing, 13(4), 653. doi: 10.3390/RS13040653
22. Tan, M., & Le, Q. V. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. In Proceedings of the 36th International Conference on Machine Learning, ICML 2019 (pp. 10691-10700).
23. Trinh, H., Haas, N., Li, Y., Otto, C., & Pankanti, S. (2012). Enhanced rail component detection and consolidation for rail track inspection. In Proceedings of IEEE Workshop on Applications of Computer Vision (pp. 289-295). doi: 10.1109/WACV.2012.6163021
24. Unser, M. (2022). Ridges, neural networks, and the Radon transform. Preprint. doi: 10.48550/ARXIV.2203.02543
25. Wang, T., Yang, F., & Tsui, K. L. (2020). Real-time detection of railway track component via one-stage deep learning networks. Sensors, 20(15), 4325. doi: 10.3390/S20154325
26. Wang, Y., Wang, L., Hu, Y. H., & Qiu, J. (2019). RailNet: A segmentation network for railroad detection. IEEE Access, 7, 143772-143779. doi: 10.1109/ACCESS.2019.2945633
27. Ye, D. H., Li, J., Chen, Q., Wachs, J., & Bouman, C. (2018). Deep learning for moving object detection and tracking from a single camera in unmanned aerial vehicles (UAVs). In IS and T International Symposium on Electronic Imaging Science and Technology. doi: 10.2352/ISSN.2470-1173.2018.10.IMAWM-466
28. Ye, T., Zhang, Z., Zhang, X., & Zhou, F. (2020). Autonomous railway traffic object detection using feature-enhanced single-shot detector. IEEE Access, 8, 145182-145193. doi: 10.1109/ACCESS.2020.3015251
29. Yu, X., Kuan, T. W., Zhang, Y., & Yan, T. (2022). YOLO v5 for SDSB distant

tiny object detection. In Proceedings of the 2022 10th International Conference on Orange Technology (ICOT). IEEE.

30. Zhang, Q., & Couloigner, I. (2007). Accurate centerline detection and line width estimation of thick lines using the Radon transform. IEEE Transactions on Image Processing, 16, 310-316. doi: 10.1109/TIP.2006.887731
31. Deng, S. et al. (2021). A global-local self-adaptive network for drone-view object detection. IEEE Transactions on Image Processing, 30, 1556-1569. doi: 10.1109/TIP.2020.3045636
32. Everingham, M., Van Gool, L., Williams, C. K., Winn, J., & Zisserman, A. (2010). The pascal visual object classes (VOC) challenge. International journal of computer vision, 88(2), 303-338.
33. Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... & Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In European conference on computer vision (pp. 740-755). Springer, Cham.
34. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., ... & Berg, A. C. (2015). Imagenet large scale visual recognition challenge. International journal of computer vision, 115(3), 211-252.
35. Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, Li Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," CVPR09, 2009.
36. Hanley, J. A., & McNeil, B. J. (1982). The meaning and use of the area under a receiver operating characteristic (ROC) curve. Radiology, 143(1), 29-36.
37. Powers, D. M. (2011). Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation.
38. Saito, T., & Rehmsmeier, M. (2015). The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. PloS one, 10(3), e0118432.
39. Gur, D., Rockette, H. E., Armfield, D. R., Blachar, A., Bogan, J. K., Brancatelli, G., ... & Yankelevitz, D. F. (2003). Prevalence effect in a laboratory environment. Radiology, 228(1), 10-14.
40. Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4700-4708).
41. Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE conference on computer vision and

pattern recognition (pp. 1251-1258).
42. Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A. A. (2017). Inception-v4, inception-resnet and the impact of residual connections on learning. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 31, No. 1).
43. 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 (pp. 770-778).
44. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. In Advances in neural information processing systems (pp. 5998-6008).
45. Hinton, G., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531.
46. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2818-2826).
47. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).
48. Zhu, Y., Yan, W.Q. Traffic sign recognition based on deep learning. Multimed Tools Appl 81, 17779–17791 (2022). https://doi.org/10.1007/s11042-022-12163-0
49. Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.
50. Zeiler, M. D., & Fergus, R. (2014). Visualizing and understanding convolutional networks. In European conference on computer vision (pp. 818-833). Springer, Cham.
51. Springenberg, J. T., Dosovitskiy, A., Brox, T., & Riedmiller, M. (2014). Striving for simplicity: The all convolutional net. arXiv preprint arXiv:1412.6806.
52. Simonyan, K., Vedaldi, A., & Zisserman, A. (2013). Deep inside convolutional networks: Visualising image classification models and saliency maps. arXiv preprint arXiv:1312.6034.

53. Zeiler, M. D., & Fergus, R. (2014). Visualizing and understanding convolutional networks. In European conference on computer vision (pp. 818-833). Springer, Cham.
54. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., & Torralba, A. (2016). Learning deep features for discriminative localization. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2921-2929).
55. Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2017). Grad-cam: Visual explanations from deep networks via gradient-based localization. In Proceedings of the IEEE international conference on computer vision (pp. 618-626).
56. Ribeiro, M. T., Singh, S., & Guestrin, C. (2016, August). "Why should i trust you?" Explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining (pp. 1135-1144).
57. Zhang, X., Yu, F., Liu, S., He, X., & Zhu, Z. (2021). Interpreting cnns with decision trees. arXiv preprint arXiv:2103.06979.
58. Huang, J. B., Singh, A., & Ahuja, N. (2015). Single image super-resolution from transformed self-exemplars. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 5197-5206).
59. Dong, C., Loy, C. C., He, K., & Tang, X. (2015). Image super-resolution using deep convolutional networks. IEEE transactions on pattern analysis and machine intelligence, 38(2), 295-307.
60. Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3431-3440).
61. Ronneberger, O., Fischer, P., & Brox, T. (2015, October). U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention (pp. 234-241). Springer, Cham.
62. 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 (pp. 770-778).
63. 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).

64. Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In Advances in neural information processing systems (pp. 91-99).
65. Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... & Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In European conference on computer vision (pp. 740-755). Springer, Cham.

LIST OF PUBLICATIONS

Paper	Author list . Title. Conference/Journal	Status
[1]	Mehrab Hosain, Rajiv Kapoor, **“A Novel APVD Steganography Technique Incorporating Pseudorandom Pixel Selection for Robust Image Security”** International Conference Com-IT-Con 2023	Accepted
[2]	Mehrab Hosain, Rajiv Kapoor “**Detection of Rail Line Track and Human Beings near the track to avoid accidents”** International Conference Com-IT-Con 2023	Accepted

9% Overall Similarity

Top sources found in the following databases:

- 4% Internet database
- 6% Publications database
- Crossref database
- Crossref Posted Content database
- 5% Submitted Works database

TOP SOURCES

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Sutao Deng, Shuai Li, Ke Xie, Wenfeng Song, Xiao Liao, Aimin Hao, Hon... Crossref	3%
2	ejurnal.its.ac.id Internet	<1%
3	dokumen.pub Internet	<1%
4	De La Salle University on 2022-06-29 Submitted works	<1%
5	Zhang Rui, Cao Ying, Bao Xintong, Teoh Teik Toe. "Image Classificatio... Crossref	<1%
6	Liverpool John Moores University on 2023-02-24 Submitted works	<1%
7	Alex Alves Bandeira. "Numerical simulations of fabric shields for bullet... Crossref	<1%
8	d-nb.info Internet	<1%

9	Swinburne University of Technology on 2018-12-09 Submitted works	<1%
10	Asia Pacific University College of Technology and Innovation (UCTI) on... Submitted works	<1%
11	umpir.ump.edu.my Internet	<1%
12	Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi. "You Only... Crossref	<1%
13	Matthew Moore, Kristen R. Walcott. "Chapter 15 Generating Human-Lik... Crossref	<1%
14	bura.brunel.ac.uk Internet	<1%
15	mediatum.ub.tum.de Internet	<1%
16	Liverpool John Moores University on 2021-06-14 Submitted works	<1%
17	Universidade Estadual de Campinas on 2021-04-26 Submitted works	<1%
18	Universiti Sains Malaysia on 2012-04-25 Submitted works	<1%
19	University of Leeds on 2021-04-17 Submitted works	<1%
20	University of Newcastle upon Tyne on 2021-04-28 Submitted works	<1%

21	University of Hull on 2020-08-03 Submitted works	<1%
22	Brisbane Grammar School on 2020-03-09 Submitted works	<1%
23	De Montfort University on 2020-05-18 Submitted works	<1%
24	University of Southampton on 2015-02-18 Submitted works	<1%
25	uir.unisa.ac.za Internet	<1%
26	scielo.br Internet	<1%
27	Lecture Notes in Computer Science, 2007. Crossref	<1%
28	UOW Malaysia KDU University College Sdn. Bhd on 2023-04-23 Submitted works	<1%
29	University of Reading on 2023-05-01 Submitted works	<1%
30	export.arxiv.org Internet	<1%
31	"Pattern Recognition and Artificial Intelligence", Springer Science and ... Crossref	<1%
32	Ahmed F Mashaly, AA Alazba. "Assessing the accuracy of ANN, ANFIS,... Crossref	<1%

33	Full Sail University on 2019-12-04 Submitted works	<1%
34	Liverpool John Moores University on 2021-08-17 Submitted works	<1%
35	Suh, Junho, and Alan Palazzolo. "3D Dynamic Model of TEHD Tilting-P... Crossref	<1%
36	University of Brighton on 2015-05-05 Submitted works	<1%
37	University of Sydney on 2021-11-21 Submitted works	<1%
38	Vedant Singh*, Vyom Unadkat, Pratik Kanani. "Intelligent Traffic Manag... Crossref	<1%
39	acervodigital.ufpr.br Internet	<1%
40	eprints.utm.my Internet	<1%
41	link.springer.com Internet	<1%
42	mis.ucd.ie Internet	<1%
43	s3-eu-west-1.amazonaws.com Internet	<1%
44	jair.org Internet	<1%

45	mdpi.com Internet	<1%
46	researchgate.net Internet	<1%