成都理工大學中英合作办学
CDUT Sino-British Collaborative Education

OXFORD BROOKES UNIVERSITY

UNDERGRADUATE PROJECT REPORT

Project Title:	**Forest Fire Detection and Severity Assessment Using Improved YOLOv5 on Satellite Imagery**
Surname:	**Qinglian**
First Name:	**Li**
Student Number:	**202118010402**
Supervisor Name:	**IRFAN ULLAH**
Module Code:	**CHC 6096**
Module Name:	**Project**
Date Submitted:	**May 5th, 2025**

Chengdu University of Technology Oxford Brookes College

Chengdu University of Technology

BSc (Single Honours) Degree Project

Programme Name: **Software Engineering**

Module No.: **CHC 6096**

Surname: Qinglian

First Name: Li

Project Title: Forest Fire Detection and Severity Assessment Using Improved YOLOv5 on Satellite Imagery

Student No.: 202118010402

Supervisor: Irfan Ullah

2ND Supervisor (if applicable): **Not Applicable**

Date submitted: **May 6th, 2025**

A report submitted as part of the requirements for the degree of BSc (Hons) in Software Engineering

At

Chengdu University of Technology Oxford Brookes College

Declaration

Student Conduct Regulations:

Please ensure you are familiar with the regulations in relation to Academic Integrity. The University takes this issue very seriously and students have been expelled or had their degrees withheld for cheating in assessment. It is important that students having difficulties with their work should seek help from their tutors rather than be tempted to use unfair means to gain marks. Students should not risk losing their degree and undermining all the work they have done towards it. You are expected to have familiarised yourself with these regulations.

https://www.brookes.ac.uk/regulations/current/appeals-complaints-and-conduct/c1-1/

Guidance on the correct use of references can be found on www.brookes.ac.uk/services/library, and also in a handout in the Library.

The full regulations may be accessed online at https://www.brookes.ac.uk/students/sirt/student-conduct/

If you do not understand what any of these terms mean, you should ask your Project Supervisor to clarify them for you.

I declare that I have read and understood Regulations C1.1.4 of the Regulations governing Academic Misconduct, and that the work I submit is fully in accordance with them.

Signature Li Qing Lian .... Date May 6th, 2025

REGULATIONS GOVERNING THE DEPOSIT AND USE OF OXFORD BROOKES UNIVERSITY MODULAR PROGRAMME PROJECTS AND DISSERTATIONS

Copies of projects/dissertations, submitted in fulfilment of Modular Programme requirements and achieving marks of 60% or above, shall normally be kept by the Oxford Brookes University Library.

I agree that this dissertation may be available for reading and photocopying in accordance with the Regulations governing the use of the Oxford Brookes University Library.

Signature Li Qing lian .... Date May 6th, 2025

Acknowledgment

This thesis was completed with the help and guidance of my Computer Science and technology research supervisor Dr. Md Irfan Ullah. From the beginning of topic selection until the completion of the entire thesis writing, he spent a great deal of time and effort to provide me with comprehensive and detailed guidance and many targeted suggestions. My mentor's profound professional knowledge, serious scientific attitude, rigorous academic spirit and tireless teaching noble ethics has all exerted a profound influence on me. He has taught me the methodology to carry out the research and to present the research works as clear as possible. It was a great privilege and honour to work and study under his guidance. I am extremely grateful for what he has offered me. I would like to express my deep and sincere gratitude to mentor for giving me the opportunity to do research and providing invaluable guidance throughout this research. I would also like to thank him for his friendship, empathy, and great sense of humour. I am extremely grateful to my parents for their love, prayers, caring and sacrifices to educating and preparing for my future. I dedicate this thesis to them.

Table of Contents

Abstract

Wildfire is a global ecological disaster, posing severe threats to ecosystem stability and biodiversity conservation. Real-time and precise monitoring of forest fires using satellite remote sensing technology has become a critical approach for mitigating environmental degradation and safeguarding human lives and property.The groundbreaking advancements in fire localization detection technology are of paramount importance, as they enable rapid and accurate identification of fire sources, significantly enhancing emergency response efficiency and providing vital decision making support for wildfire containment. The paper proposed an advanced object detection framework named YOLOv5s-Fire based on the lightweight YOLOv5s architecture which is specifically designed for the detection of wildfire locations in satellite imagery. The proposed model incorporates GhostConv layers for computational efficiency and CBAM attention mechanisms to enhance feature representation in wildfire patterns and combine the C3Ghost, SPPFire module. The architecture achieves 33% reduces parameters while maintaining detection accuracy through strategic channel compression. Meanwhile, a robust wildfire detection model requires precise and efficient localization of fire positions within forested satellite scenes. The 400 satellite images were filtered from existing datasets, with 75% allocated for training and 25% for validation. Experimental results demonstrate that YOLOv5s-Fire is capable of successfully detecting wildfires and exhibits superior performance compared to YOLOv5 for early fire detection, with a 96.7% precision and the $\boldsymbol{mAP_{0.5}}$ is 1.7% higher than YOLOv5s. The proposed model provides a lightweight and efficient solution suitable for deployment in resource constrained wildfire monitoring systems, which is expected to achieve earlier fire detection and early warning.

Keywords: forest fire; YOLOv5 object detection; satellite image; wildfire

Abbreviations

Abbreviation	Definition
YOLOv5s	You Only Look Once version 5 small
CBAM	Convolutional Block Attention Module
SPPFire	Spatial Pyramid Pooling Fire
GhostConv	Ghost Convolution
C3Ghost	Cross-stage partial network with 3 GhostConv blocks
nc	Number of classes
lr	Learning Rate
IoU	Intersection over Union
mAP	mean Average Precision
$mAP_{0.5}$	mAP at IoU threshold 0.5
$mAP_{0.5:0.95}$	Average mAP over IoU thresholds 0.5 to 0.95
$CIoU$	Complete Intersection over Union
BCE	Binary Cross Entropy
Box loss	bounding box regression
precision	Ratio of true positives to all predicted positives
recall	Ratio of true positives to all actual positives
CNN	Convolutional Neural Network
FPN	Feature Pyramid Network
PAN	Path Aggregation Network
NMS	Non-Maximum Suppression
GDPR	General Data Protection Regulation
IEEE	Institute of Electrical and Electronics Engineers
BCS	British Computer Society
ACM	Association for Computing Machinery

Glossary

YOLOv5s

It’s a lightweight variant of YOLOv5 object detection model, made faster for speed and real-time applications.

CBAM (Convolutional Block Attention Module)

It is an attention mechanism which first applies channel attention and then applies spatial attention for feature representation in convolutional neural networks.

SPPFire (Spatial Pyramid Pooling Fire)

SPPFire is a modified spatial pyramid pooling layer specifically designed for multi-scale fire characteristic extraction in satellite imagery.

GhostConv (Ghost Convolution)

The lightweight convolution operation that generates more feature maps using cheap linear transformations to reduce computational costs.

C3Ghost

The cross-stage partial network module incorporating three GhostConv blocks for efficient feature reuse in the backbone network.

mAP (mean Average Precision)

The metric evaluating object detection model performance by averaging precision values across all recall levels, with thresholds at IoU=0.5 (mAP 0.5) and IoU=0.5-0.95 (mAP 0.5:0.95).

IoU (Intersection over Union)

Intersection over union overlap of the intersection over union overlap between predicted and ground truth boxes.

Feature Pyramid Network (FPN)

The A neural network architecture that builds high-level semantic feature maps at multiple scales for object detection.

Path Aggregation Network (PAN)

It is a network structure that enhances feature fusion by bottom-up path augmentation to improve localization accuracy.

Non-Maximum Suppression (NMS)

A post-processing technique that eliminates redundant bounding box predictions by selecting the highest-scoring detection among overlapping proposals.

Channel Compression

A model optimization technique that reduces the number of feature channels while maintaining detection accuracy through strategic parameter pruning.

Class Imbalance

A data distribution problem where the number of samples in different classes varies significantly, potentially biasing model training.

Edge Devices

Resource-constrained computing devices deployed at the network edge capable of running lightweight AI models.

Chapter 1 Introduction

1.1 Background

Forests as a crucial component for maintaining the ecological balance of the Earth [1], [2], encompass abundant natural resources [3], [4] and offer the environment requisite for human survival [5]. Nevertheless, unforeseeable human factors and natural disasters have triggered the rapid and uncontrollable spread of forest fires [1], [3], [6], posing a significant threat to ecosystems and human society, including direct losses of life and property, as well as long-term resource pollution issues [7], [8], [9]. As indicated by statistics from the Global Wildfire Information System, the average burned area of each wildfire amounts to as high as 20 hectares, and over 25,000 hectares of forest are devastated by wildfires annually [10]. Hence, detecting the location and severity of forest fires rapidly and precisely is an essential measure to alleviate their detrimental effects.

Although traditional monitoring methods, such as ground sensors [11] and drone patrols [12], [13], have reduced the incidence of forest fires to some extent, they are still limited by factors such as response speed and coverage [2], [6], [12]. With the rapid development of remote sensing technology, such as high-coverage datasets provided by Moderate Resolution Imaging Spectroradiometer (MODIS) satellites offer a unique perspective for fire monitoring, solving the dilemma of quickly covering large areas and providing valuable data resources for early detection and dynamic tracking of fires [13], [14]. However, due to the relatively low spatial resolution of imagery and the diverse manifestations of areas affected by fires, it is challenging to directly determine the exact location and severity of fires based on satellite imagery data alone. The application of machine learning techniques, in particular Convolutional Neural Networks (CNN), has yielded significant outcomes in the domains of image classification and target detection [15], which has brought new opportunities for remote sensing fire detection [16], [17].

In this project, an improved YOLOv5-based deep learning framework is proposed to be trained to analyze satellite data to automatically identify areas where forest fires are occurring, estimate burned areas and assess fire severity. This approach achieves a faster and more accurate detection of forest fires, which enables rescuers to quickly understand the size of the fire and its potential impacts and formulates a more effective emergency response strategy. The rest of the proposal is structured as follows: In Section 2, the literature is reviewed and a comparative analysis of existing approaches is provided. Section 3 introduce the techniques required for the project research and the processing of dataset. Section 4 elaborates the implementation details and explores the experimental results in depth, and compares the efficiency of the proposed model with other current wildfire detection models. Section 5 presents the project management plan and associated risk analysis, and a brief overview is given in Section 6.

1.2 Aim

The aim of this proposed project is to develop a deep learning framework based on YOLOv5 to quickly identify the location of wildfire.

1.3 Objectives

There are nine objectives proposed of the project.

(1) Review the relevant literature and evaluate existing forest fire monitoring models.

(2) Select and collect appropriate data

(3) Preprocess the image data.

(4) Split the data into training, validation, and test sets.

(5) Design an improved YOLOv5 model suitable for satellite wildfire identification.

(6) Use the training set to train the constructed model.

(7) Use verification sets to evaluate the performance of the trained model on new data.

(8) Use test sets to test the accuracy of the model on the forest fire detection.

(9) Presenting the results of the research to the audience.

1.4 Project Overview

1.4.1 Scope

The purpose of this proposed project is to develop a improved deep learning framework based on YOLOv5, which aims to facilitate rapid detection of forest wildfire area. Through the application of deep learning techniques, it is possible to enhance the accuracy and responsiveness of the forest fire monitoring system, thereby facilitating the implementation of timely emergency management measures during the initial stages of a fire, which can mitigate the impact of ecological damage and economic losses. Furthermore, this research contributes to more efficient wildfire monitoring and response strategies and provides referable cases for advancing academic research in the field of remote sensing image processing.

1.4.2 Audience

This proposed project will prove beneficial to forest managers, emergency rescue teams and the general public. Primarily, forest managers can utilize the automated systems deployed for the monitoring and management of forest resources, particularly during periods of elevated fire risk, thereby enabling the provision of early warnings and the implementation of preventive measures. The swift and precise dissemination of information regarding the location and extent of fires can facilitate the prompt response of rescue teams, thereby reducing the loss of life and property. Through the real-time issuance of fire warning notifications, the public can access timely information on the risks associated with forest fires, enabling them to safeguard their lives and property, while also fostering awareness of forest fire prevention measures.

Chapter 2 Background Review

This proposed project investigates and compares from the state of the art techniques for forest fire detection using deep learning methods. In [3], Khan et al. put forward a transfer learning approach based on VGG19 for the detection of forest fires. The experiment utilized 80% of the data in the DeepFire dataset for training and attained an accuracy of 95.72% in the 20% test set, where there were 950 images each for the fire class and the non-fire class. Seydi et al. [1] employed a Landsat-8 image set for forest fire detection and proposed a deep learning framework Fire-Net integrated by YOLOv5 and U-Net network, which improved the accuracy of forest small fire detection under different conditions. Yuan et al. [5] employed the 85% FLAME dataset as a training set to enable the model to acquire global context information by incorporating a multi-head self-attention (MSA) module before each YOLO header. Experimental results have proven that this approach can markedly enhance the efficacy of forest fire detection across different scales while preserving the real-time performance of fire detection. In contrast to the approach taken in [5], [18] considered the diversity of forest fire scenarios and integrated YOLOv5 with EfficientNet, significantly reducing the false alarm rate in forest fire detection by learning global information. The model demonstrated a high level of accuracy in recognizing fire images, achieving 99.6% accuracy on 476 images and 99.7% accuracy on 676 images. Kang et al. [19] proposed a deep learning model based on geostationary satellite Himawari-8 AHI data, which employed temporal and spatial information features to markedly reduce the detection delay of forest fires. [20] proposed a Forest Smoke-Fire Net (FSF-Net) model based on Mask R-CNN, which combining MODIS remote sensing images with regional dynamic brightness temperature thresholds. The study demonstrates that the model can effectively circumvent the issue of erroneous identification and missed detection resulting from interference such as cloud and fog, and significantly improve the precision and dependability of forest smoke detection. Yang et al. [21] introduced a novel Squeeze-Excitation Spatial Multi-Scale Transformer Learning (SESMTML) algorithm that integrates deep learning with remote sensing imagery to tackle the spatial and temporal challenges in forest fire risk prediction models and the lack of universality due to regional inconsistencies. The comparison of the related studies is shown in **Table 1**.

Research	**Model**	**DataSet**	**Performance Index**

Khan et al. [3]	VGG19	DeepFire	Accuracy = 98.89% Precision = 95.72%
Seydi et al. [1]	Fire-Net (YOLOv5 + U-Net)	Landsat-8	Overall Accuracy = 97.35% Precision = 93.49%
Yuan et al. [5]	YOLOv5 + Transformer	FLAME, self-built fire dataset	Accuracy = 93.25% Precision = 92.85%
Xu et al. [18]	YOLOv5 + EfficientNet	BowFire, FD-dataset, ForestryImages, VisiFire	Accuracy = 99.6% Average Precision = 85.5%
Kang et al. [19]	CNN + RF	Himawari-8 AHI	Overall Accuracy = 98% Precision = 91%
Ding et al. [20]	FSF-Net (Mask R-CNN)	MODIS_Smoke_ FPT dataset	Accuracy = 89.12%
Yang et al. [21]	SESMTML (CNN + Transformer)	FireRisk	Overall Accuracy = 83.18% Precision = 83.05%

Table 1 Comparison of different research

Chapter 3 Methodology

3.1 Approach

3.1.1 YOLOv5

The YOLOv5 algorithm is characterized by multi-scale detection and lightweight target localization, and is mainly composed of three parts: Backbone, Neck, and Head [22]. The YOLOv5 structure is shown in **Fig. 1**. The Backbone network as the core of feature extraction in YOLOv5, comprises BottleneckCSP and Focus modules. It extracts features from the input image through multiple convolutional layers and pooling layers, and fuses feature maps of different scales via cross-layer connections and channel compression, ultimately outputting feature maps with semantic information.

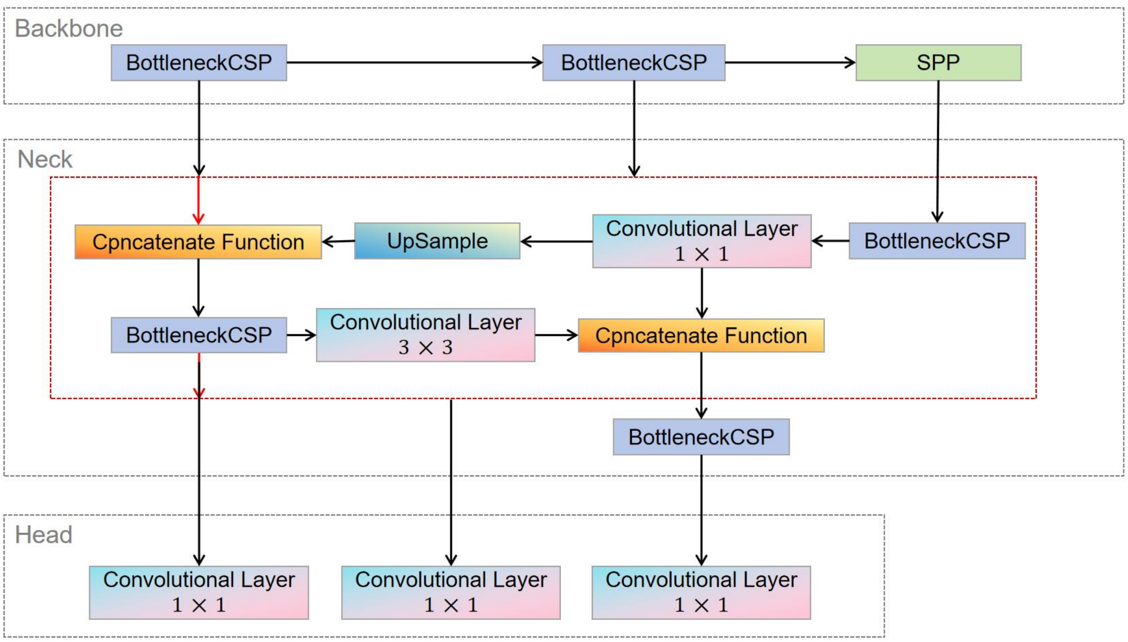

Fig. 1 YOLOv5 Structure

In the task of forest fire detection, detection speed and accuracy are of vital importance, and the compact model size determines the inference efficiency on resource-scarce edge devices. BottleneckCSP draws on the CSPNet network model and consists of three convolutional layers and multiple Res unit modules Concate, which enhances the learning performance of the entire convolutional neural network while significantly reducing the computational cost [22], [23]. The Focus module conducts slice operations on the image, expands the input channels to four times the original, and acquires a downsampled feature map through a single convolution. This achieves downsampling while reducing the computational cost and improving the training speed of the model.

In the Neck, the FPN and PAN structures are adopted, and the CSP2 structure inspired by CSPNet is introduced to enhance the feature fusion capability of the network [22]. The CSP module first divides the feature map of the base layer into two parts and then combines them through a cross-stage hierarchical structure, reducing the computational cost while guaranteeing the detection accuracy [23]. The FPN structure transmits and fuses the high-level semantic features through upsampling in a top-down manner; while the PAN structure transmits the location features of the lower layers through downsampling in a bottom-up manner [22]. The combination of the conventional FPN layer and the PAN structure fuses the extracted semantic features and location features, and simultaneously fuses the features of the main body and the detection layer, enabling the model to obtain more abundant feature information and significantly enhance its detection ability. Forest fires typically progress from small-scale fires (ground fires) to medium-scale fires (trunk fires) and then to large-scale fires (canopy fires). Multi-scale detection of YOLOv5 ensures that the model can track the size changes during the evolution of the fire. The Spatial Pyramid Pooling (SPP) module applies pooling operations at different scales $(1 \times 1, 5 \times 5, 9 \times 9, 13 \times 13)$ for multi-scale fusion, ensuring that the output features are invariant and multi-scale and avoiding the loss of some information [23]. In the Head network structure, as shown in Equation (1), YOLOv5 employs the CIoU Loss as the loss function for the bounding box [22], [23]. By adding an influencing factor to the DIoU Loss, the scale information of the aspect ratio of the bounding box is considered.

$$\mathrm{CIOU\_Loss} = 1 - \mathrm{CIOU} = 1 - (\mathrm{IOU} - \frac{\mathrm{Distance\_2}^2}{\mathrm{Distance\_C}^2} - \frac{\upsilon^2}{(1 - \mathrm{IOU}) + \upsilon}) \tag{1}$$

3.1.2 DataSet Selection

This proposed project intends to use the Wildfire Prediction Dataset (Satellite Images) from Kaggle for the object detection problem of wildfire images. The dataset is derived from satellite images of areas that have previously experienced wildfires in Canada and is divided into two classes: fire and no-fire. The fire class contains forest and mountain regions or images with invisible black flames or gray smoke clouds. In contrast, the no-fire class contains images of forests and cities from different angles. The random samples of images both classes in the Wildfire dataset are shown in **Fig. 2**.

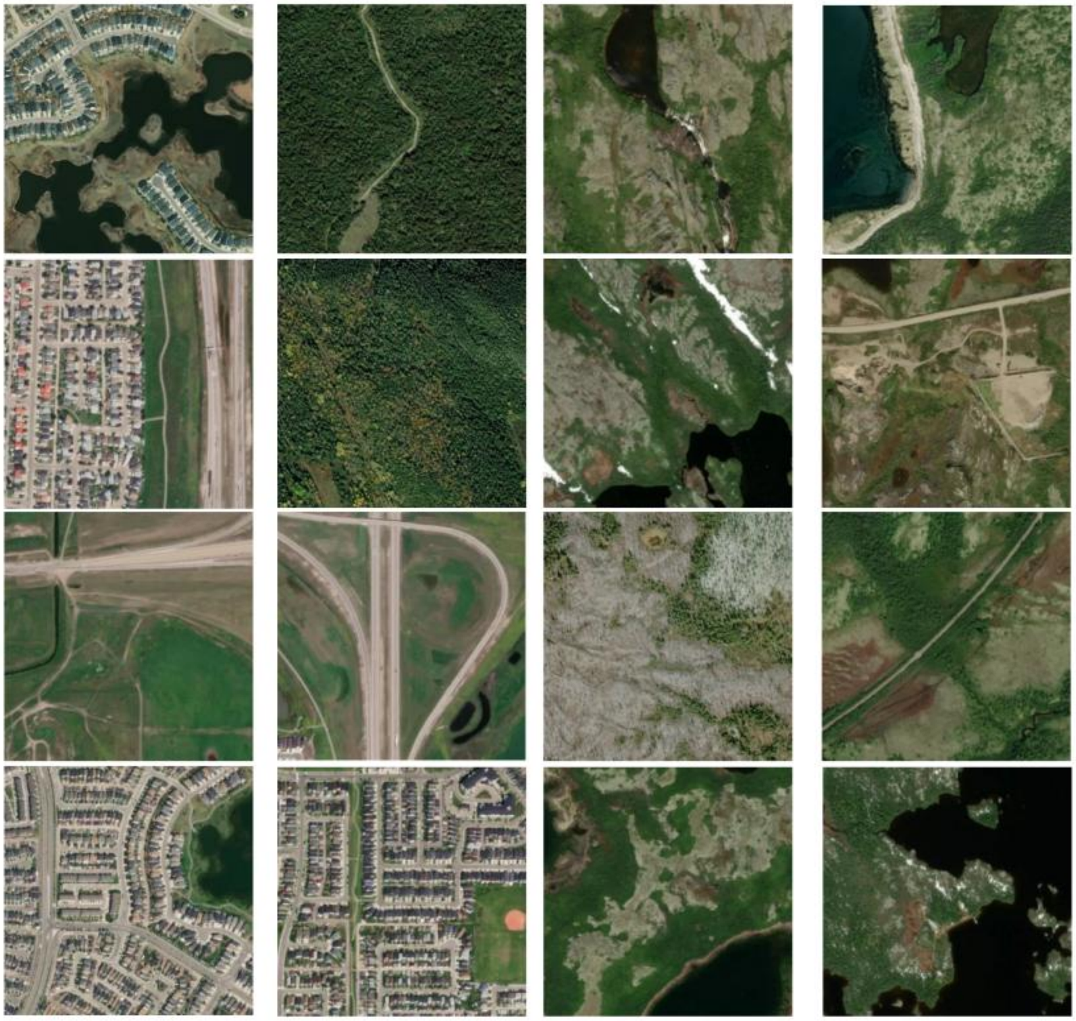

Fig. 2 Images of wildfire and no-wildfire classes

There are a total of 42,850 images in the Wildfire dataset, of which 22,710 are fire examples and the remaining 20,140 belong to the no-fire class, as shown in **Table 2**. All images have an initial resolution of 350 × 350 pixels.

DataSet	Training	Testing	Validation	Total
wildfire	15898	3406	3406	22710
No-wildfire	14098	3021	3021	20140
Total	400	6427	6427	42850

Table 2 Dataset distribution

3.1.3 Data Preparation

Based on the original Wildfire data set, this project constructs a high-quality sample set through manual screening. Screening criteria included:

(1) Image clarity: exclude blurred or low-resolution samples.

(2) Label reliability: wildfire images should clearly contain fire or smoke features and no-wildfire images should have no relevant traces.

(3) Scene coverage: it covers diverse backgrounds such as forests, mountains and cities.

After filtering, a total of 400 images were selected, consisting of 200 wildfire samples and 200 non-wildfire samples. The dataset was deliberately balanced in terms of class distribution to mitigate training bias and facilitate more stable convergence during optimization. The balanced composition ensures that the model can effectively learn to distinguish between wildfire and no-wildfire scenes, promoting fair and reliable evaluation across all sample types.

3.1.4 Data Splitting

The filtered dataset was subsequently partitioned into a training set and a validation set using a randomized 75%:25% split ratio, yielding 300 training images and 100 validation images, as shown in **Table 3**. The splitting process was performed with stratified sampling to ensure that the wildfire and no-wildfire remained consistent across subsets.

DataSet	Training	Validation	Total
wildfire	150	50	200
No-wildfire	150	50	200
Total	300	100	400

Table 3 Dataset splitting

To independently evaluate the model's generalization performance, a separate test set comprising 50 images was sampled from the unfiltered portion of the original dataset. In order to prioritize the assessment of the model's sensitivity and accuracy in detecting fire-related events, the test set was deliberately constructed to include 40 wildfire images and 10 non-wildfire images, as detailed in **Table 4**. This test configuration not only reflects the critical importance of accurate wildfire detection in emergency scenarios but also serves to validate the model's practical applicability under conditions of class imbalance and visual variability.

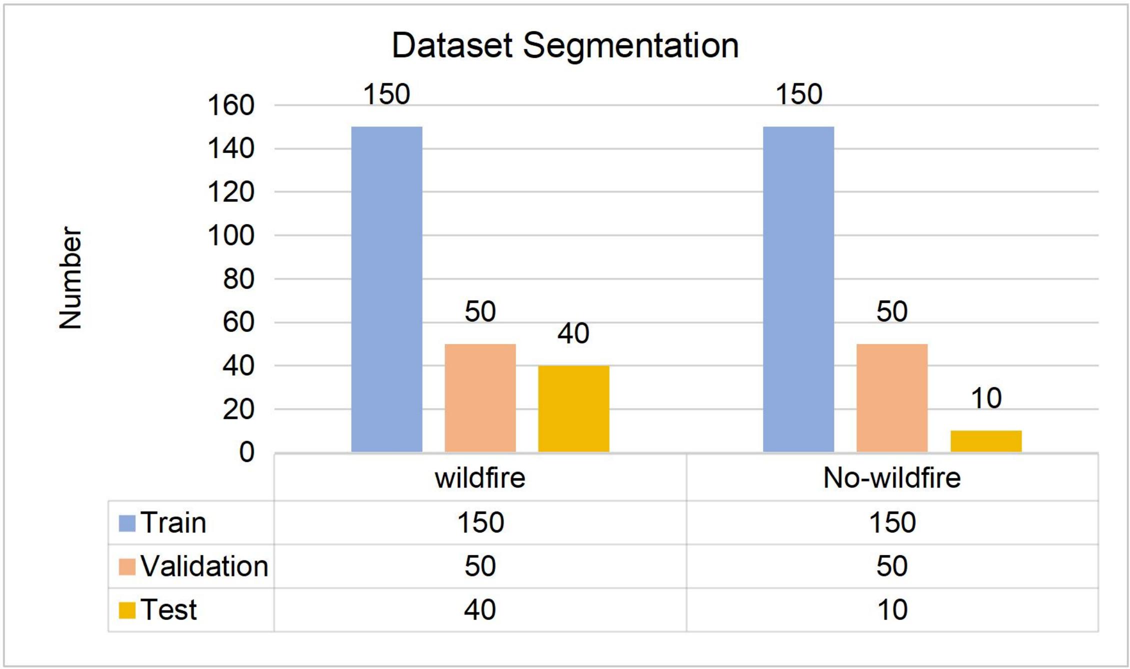

Table 4. Dataset Segmentation

3.1.4 Data Preprocessing

To ensure high-quality input for model training and evaluation, we implemented a structured data preprocessing workflow encompassing both wildfire and non-wildfire imagery. For wildfire samples, we employed the open-source annotation tool LabelImg to manually annotate visible fire regions with bounding boxes. As shown in **Fig. 3**, the annotated fire images are displayed. The annotations were saved in the YOLO format, which records object class and normalized bounding box coordinates, ensuring compatibility with real-time object detection frameworks. This format is lightweight and well-suited for real-time object detection tasks. For non-wildfire images, which inherently contain no fire regions, we generated corresponding empty label files to denote the absence of any target objects. These images serve as negative samples, enabling the model to distinguish between fire and non-fire scenes more effectively. This annotation strategy enables a clear distinction between positive and negative samples, promotes robust feature learning, and supports effective training in the presence of class imbalance, which is critical for minimizing false positives and enhancing the robustness of the detection system in diverse real-world environments.

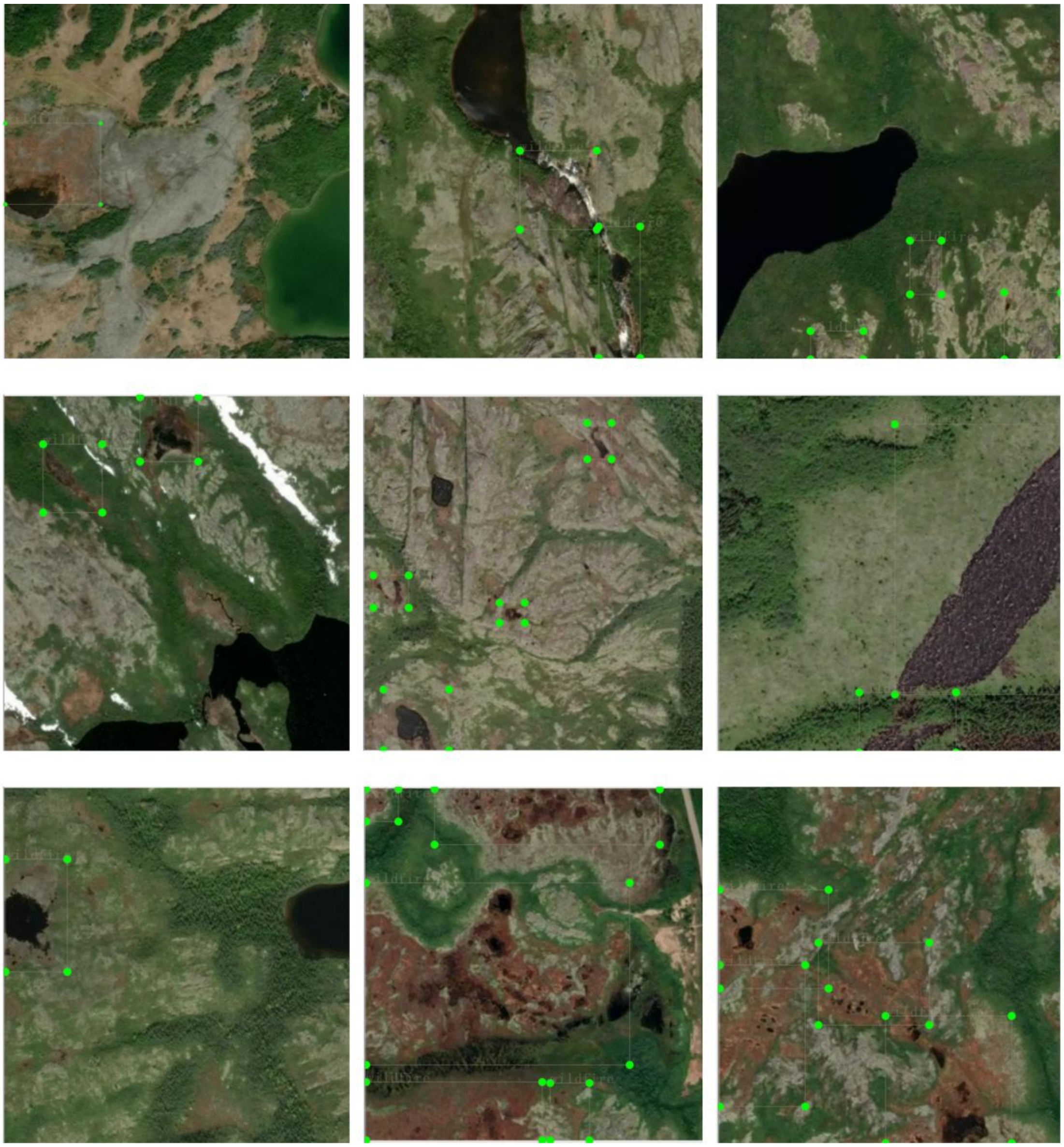

Fig. 3 Random samples after wildfire image labeling

3.1.5 Model Construction and Optimization

This project utilizes the YOLOv5 model to achieve efficient detection of “wildfire” targets. As a lightweight and high-performance object detection model, YOLOv5s strikes an excellent balance between speed and accuracy, making it particularly suitable for processing large-scale satellite image datasets. Based on the characteristics of the data and task requirements, the following optimizations have been made to the YOLOv5s model:

a) Data Augmentation Strategies: Techniques such as Mosaic data augmentation, random cropping, color jittering, and random flipping were employed to effectively enhance data diversity, enabling the model to better learn features in complex scenes and improve generalization capabilities.

b) Hyper-parameter Optimization: Key YOLOv5s hyperparameters were fine-tuned based on the distribution characteristics of the training data, including learning rate, batch size, IoU threshold, and Non-Maximum Suppression (NMS) strategy, thereby improving detection accuracy and stability.

c) Loss Function Improvements: To improve the detection accuracy of the proposed YOLOv5s-Fire model, the Complete Intersection over Union (CIoU) loss was employed to enhance the precision of bounding box regression by considering overlap area, distance, and aspect ratio simultaneously. In addition, Focal Loss was integrated into the classification branch to mitigate the negative effects of class imbalance, ensuring the model maintains robustness when facing a high proportion of background or easy negative samples in wildfire detection tasks.

d) Transfer Learning: The weights of YOLOv5 pre-trained on COCO dataset are used as the initial parameters of the model, and the convergence speed and detection performance of the model on remote sensing images are significantly improved by transferring the existing general object detection features.

These optimization measures ensure that the model is well-suited to the characteristics of satellite imagery for wildfire and non-wildfire targets, achieving a superior balance between detection accuracy and computational efficiency, and laying a solid foundation for subsequent tasks.

3.2 Technology

The techniques used to implement these projects are shown in **Table 3**.

Software	Framework	PyTorch2.1.0
	Language	Python 3.x
	Operating System	Windows 11

	System development	PyCharm 2024.3.4
	Version management plan	Baidu Cloud
		GitHub repository
Hardware	GPU	NVIDIA GeForce RTX 4060 Laptop GPU
	CPU	The 14th Gen Intel® Core™ i9-14900HX CPU 32 cores
	Memory	32GB
	SSD	1TB

Table 5 Tools and techniques for development

The models discussed in this study were implemented using Python in a Lenven Windows 11 system with 32 GB of DDR5. The system features 14th Gen Intel® Core™ i9-14900H processor with a base clock speed of 2.20 GHz and a maximum turbo frequency of 5.8 GHz, providing substantial processing power for the model training and evaluation processes. The system is equipped with an NVIDIA GeForce RTX 4060 standalone graphics card, providing efficient performance.

3.3 Testing and Evaluation Plan

The process of testing and evaluation is shown in **Fig. 4**.

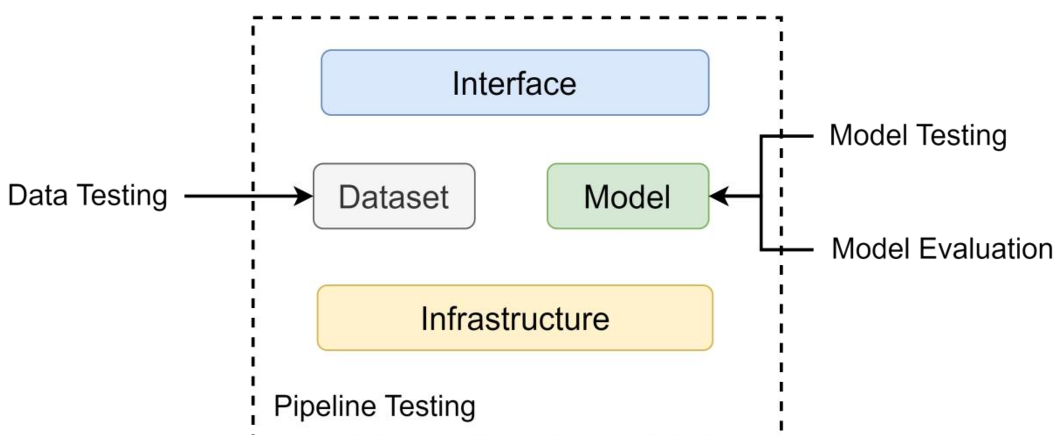

Fig. 4 Testing and Evaluation Process

3.3.1 Dataset Testing Plan

The testing plan of the dataset is shown in **Table 4**.

Type	Plan
Image integrity check	1. Verify that the picture file is corrupt, such as read failure or incomplete data.
	2. Check whether the resolution of the images is uniform, and adjust all the images to the model to the standard size required for training
	3. Check whether the range of pixel values meets the requirements of the model and is correctly normalized
Data enhancement test	1. Test whether you can successfully use different enhancement methods such as flipping, cropping, rotation, blur, and brightness adjustment
	2. Verify that the enhanced pictures retain the integrity of the fire target without producing artifacts or noise.
Data distribution analysis	1. Check whether the distribution of various data is balanced.
	2. To check whether the distribution of pictures in the training, validation and test sets were consistent, including similar proportions of fire severity and background.
	The size and location of fire detection targets were counted to ensure diversity during model training.

Table 6 Testing Plan of Dataset

3.3.2 Model Testing Plan

a) Pre-train Testing

- Verify that the model inputs are correct, such as input image size, format, and normalization range.

- Test whether the model architecture is properly defined for the target task. Whether the classification head and bounding box predictions are correctly connected.

b) Post-train Testing

- Invariant Tests: Evaluate the robustness of the model under meaningless input perturbations when rotation or brightness adjustment, and ensure its performance remains stable.
- Directional Tests: Assess whether the model's output aligns with expectations under specific input changes, such as predicting a higher severity score when an image with a larger fire area is input.
- Minimum Functional Tests: Test the model's basic functionality on a small-scale dataset to ensure it can make reasonable predictions for given known labels.

3.3.3 Model Evaluation Strategy

This research will use seven metrics to evaluate the difference in the detection of several image classes in the same experimental environment. These metrics include Precision, Recall, box_loss, obj_loss, $mAP_{0.5}$, $mAP_{0.5:0.95}$. Each of the metrics is mathematically expressed as follows:

The True Positive (TP) represents that the true class of the sample is positive and the model correctly identifies it as positive. False Negative (FN) represents that the true class of a sample is positive, but the model incorrectly identifies it as negative. False Positive (FP) represents that the true class of a sample is negative, but the model incorrectly identifies it as positive. True Negative(TN) represents that the true class of the sample is negative and the model correctly identifies it as negative [29], [30], [31].

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

Precision is an indicator to evaluate the accuracy of the model in predicting positive class samples, which indicates the proportion of samples predicted by the model as positive class that are actually positive class [29], [31]. The higher the accuracy, the stronger the prediction ability of the model for the positive class, and the more accurate the classification result.

$$Recall\ (SE) = \frac{TP}{TP + FN} = \frac{TP}{P} \quad (3)$$

Recall is a measure to evaluate the performance of a model in identifying positive class samples, and it indicates the proportion of all actual positive class samples that the model can identify[31].

The box_loss measures the positional discrepancy between the predicted bounding boxes and the ground truth[32]. It is computed as the average difference between 1 and the IoU for all predicted-ground truth box pairs:

$$box\_loss = \frac{1}{N}\sum_{i=1}^{N}(1 - IoU(B_i^{pred},\ B_i^{gt})\,) \tag{4}$$

where N denotes the number of bounding boxes. YOLOv5 uses CIoU, which considers IoU, center distance, and aspect ratio[22], [23]。

The obj_loss evaluates how accurately the model predicts the presence of objects[33]. It is based on the BCE loss function:

$$obj\_loss = -\ \frac{1}{N}\sum_{i=1}^{N}[y_i \cdot log(p_i) + (1 -\ y_i) \cdot log(1 - p_i)] \tag{5}$$

where y is the ground truth,1 for object, 0 otherwise, and p is the predicted objectness score. The smaller the obj_loss, the more accurate the model is in determining whether there is a target in the image.

The $mAP_{0.5}$ evaluates the detection accuracy when the IoU between predicted and ground truth boxes is at least 0.5 [34]. It is calculated as the mean of AP across all classes:

$$mAP_{0.5} =\ \frac{1}{C}\sum_{c=1}^{C} AP_c(IoU \geq 0.5) \tag{6}$$

$mAP_{0.5:0.95}$ is a stricter metric introduced by the COCO dataset. It averages the APs calculated at IoU thresholds ranging from 0.5 to 0.95 with a step size of 0.05[35]:

$$mAP_{0.5:0.95} = \frac{1}{10}\sum_{t=0.5}^{0.95}[\frac{1}{C}\sum_{c=1}^{C} AP_c(IoU \geq t)\,], t \in \{0.5, 0.55, \ldots, 0.95\} \tag{7}$$

This metric provides a comprehensive view of the detector’s performance under varying localization precision.

The cross-entropy loss function is commonly employed in multi-class classification tasks, as it quantifies the discrepancy between the predicted probability distribution and the true label distribution. During model training, the cross-entropy loss function evaluates the accuracy of the model’s predicted probabilities, guiding the optimization of model parameters and gradually enhancing performance in multi-class classification. The formula for this loss function is as follows:

$$Loss = -\sum_{i=1}^{C} y_i log(p_i) \quad (8)$$

Here, C is the number of classes, y_i represents the true one-hot encoding, where $y_i = 1$ for the correct class and $y_i = 0$ for other classes. The p_i is the predicted probability of class i, typically output by the $\mathrm{softmax}$ function. This loss function effectively penalizes incorrect predictions, particularly when the model assigns a high probability to a class that does not match the actual label. In such cases, the loss increases significantly, providing a strong incentive for the model to adjust its predictions. As a result, the cross-entropy loss function steers the model toward accurate class predictions during training, ultimately improving classification performance.

3.3.4 Pipeline Testing Plan

The testing plan of the pipeline shown in **Table 5**.

Type	Testing Plan
Data Handling	Verify that the image preprocessing module can correctly handle all images, including cropping, scaling, normalization, and data enhancement steps.
	Random sampled images were checked manually to ensure that the processing results were as expected.
Model Training & Testing	When training the model, check whether the picture batch loading is stable, and whether the labels are correctly loaded into the corresponding picture.
	When testing the model, test whether the output result of the model is consistent when entering a single picture and a batch picture.
Integration testing	The complete process from the original picture to the detection output, verify that all modules in the pipeline are seamlessly connected to avoid data transfer errors.

Table 7 Testing Plan of Pipeline

3.4 Project Version Management

As shown in the table below, three versions of the project are expected.

Version	Description
1.0	Implement YOLOv5 models
2.0	Implement the improved YOLOv5 model
3.0	Optimized theimproved YOLOv5 model

Table 8 Version management

To avoid confusion or accidental loss of project documentation and code, the following resources are used to effectively manage all documentation:

- Baidu Cloud, where I sync and store project changes in order to revert back to the correct version in a timely fashion.

  URL: https://pan.baidu.com/s/1Q67PsmEjM_C2G1W-kq2p_Q?pwd=o52s.

- GitHub repository, where I will upload the code and related work for the confirmed version model as a backup.

  URL: https://github.com/YQJane/L6C4-202118010402-Project.

Chapter 4 Implementation and Results

This section shows the implementation of YOLOv5s-Fire and the test results of different YOLOv5 versions on the dataset. The proposed model is compared with YOLOv5s, YOLOv5n, YOLOv5m, YOLOv5l using the same training settings, environment, and datasets.

4.1 Proposed Model

As demonstrated in **Fig. 5**, the YOLOv5s-Fire model exhibits an optimized architecture that integrates lightweight model with attention mechanisms for efficient object detection. The proposed model is composed of two primary components: a backbone network, which is responsible for feature extraction, and a detection head, which is responsible for multi-scale prediction.

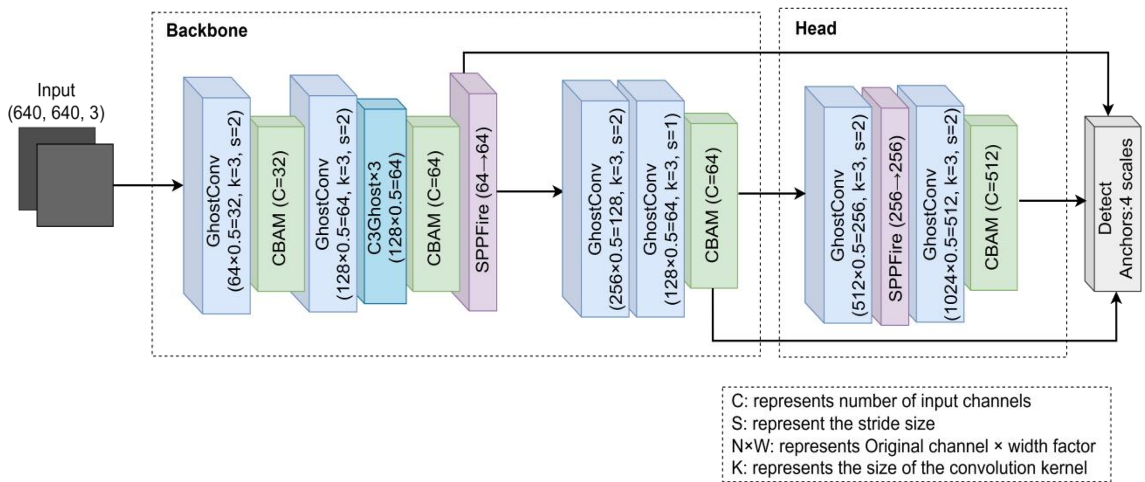

Fig. 5 The architectural of the YOLOv5s-Fire model

This object detection architecture achieves an efficiency-accuracy balance through multi-stage optimizations. The backbone network employs GhostConv modules in initial layers for feature extraction, utilizing decoupled convolution strategies to reduce computational costs while preserving rich feature information. The 3×3 kernel with stride 2 effectively compresses spatial dimensions, laying the foundation for subsequent processing. The cascaded channel attention mechanism and spatial attention module enable dynamic focus on critical feature regions, significantly enhancing small object detection capabilities. Deep network structures adopt improved C3Ghost units, which stack three GhostConv layers while retaining cross-layer residual connections. This design inherits the advantages of traditional residual architectures while leveraging GhostConv's parameter compression, reducing computational load per module by approximately 37%.

The detection head implements a progressive channel reduction strategy combined with dynamic upsampling, gradually restoring feature map resolution while controlling parameter scale. The output stage maintains YOLOv5's multi-scale detection mechanism, preserving anchor box configurations to ensure pipeline compatibility. Strategic embedding of attention modules during feature fusion effectively mitigates missed detection issues common in lightweight models.

4.2 Overall architecture process structure

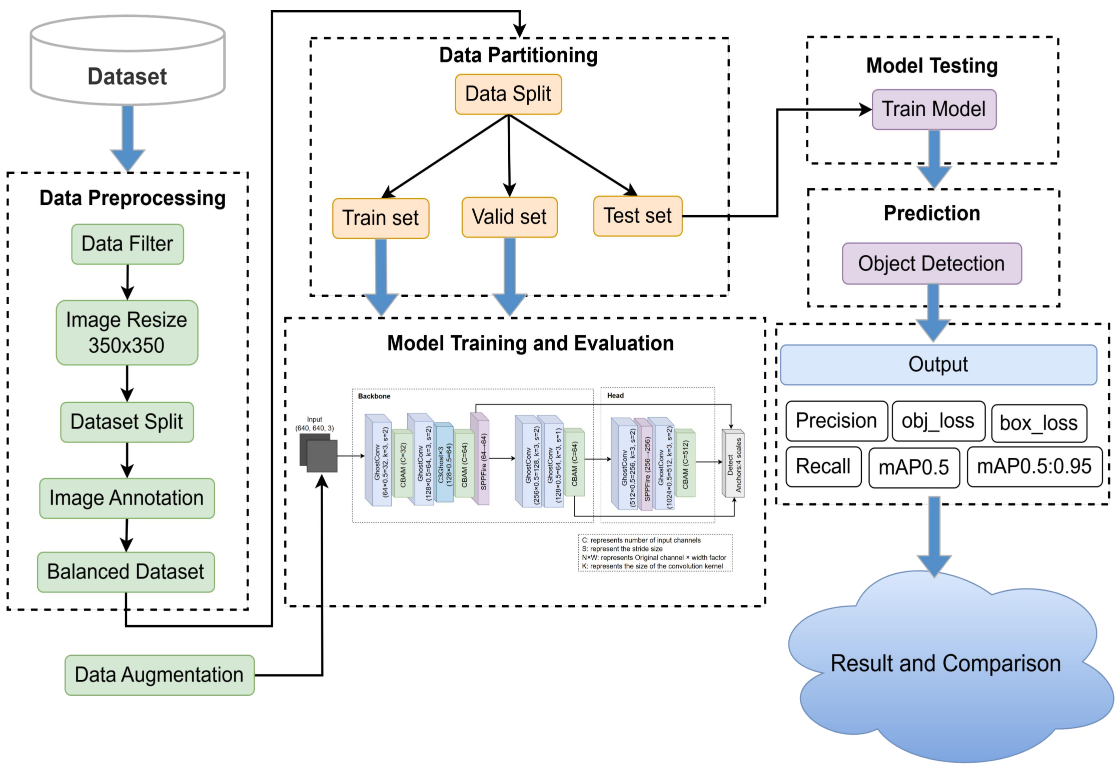

Fig. 6 The overview of the overall architecture process structure

The **Fig. 6** illustrates the overall process architecture of the proposed model in this study. Data preprocessing mainly includes data filtering, dataset segmentation and image annotation. The dataset is split into training and validation sets. Before the model training, the images were augmented, and 50 satellite images were randomly selected as the training set during the training process. The trained model is evaluated on the test set to detect wildfires in various scenes of satellite imagery. Model performance is measured and compared through the six metrics shown in the figure.

4.3 Experimental Results

4.3.1 YOLOv5s

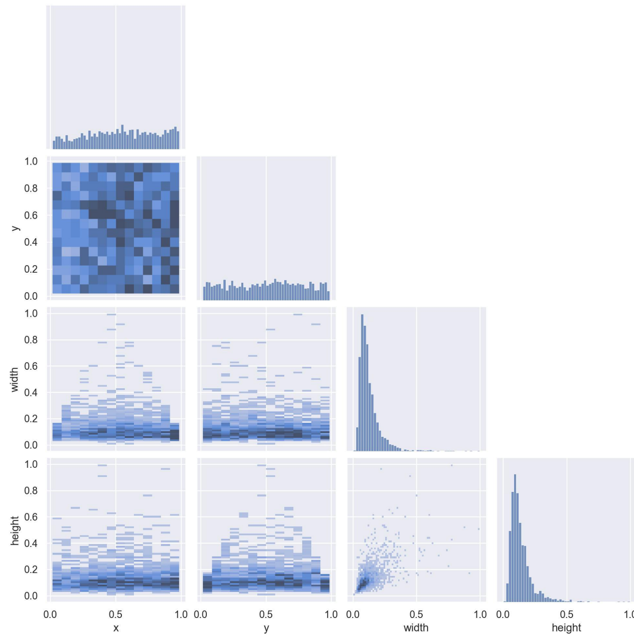

Fig. 7 The matrix diagram of the four variable joint distribution and correlation analysis of YOLOv5s

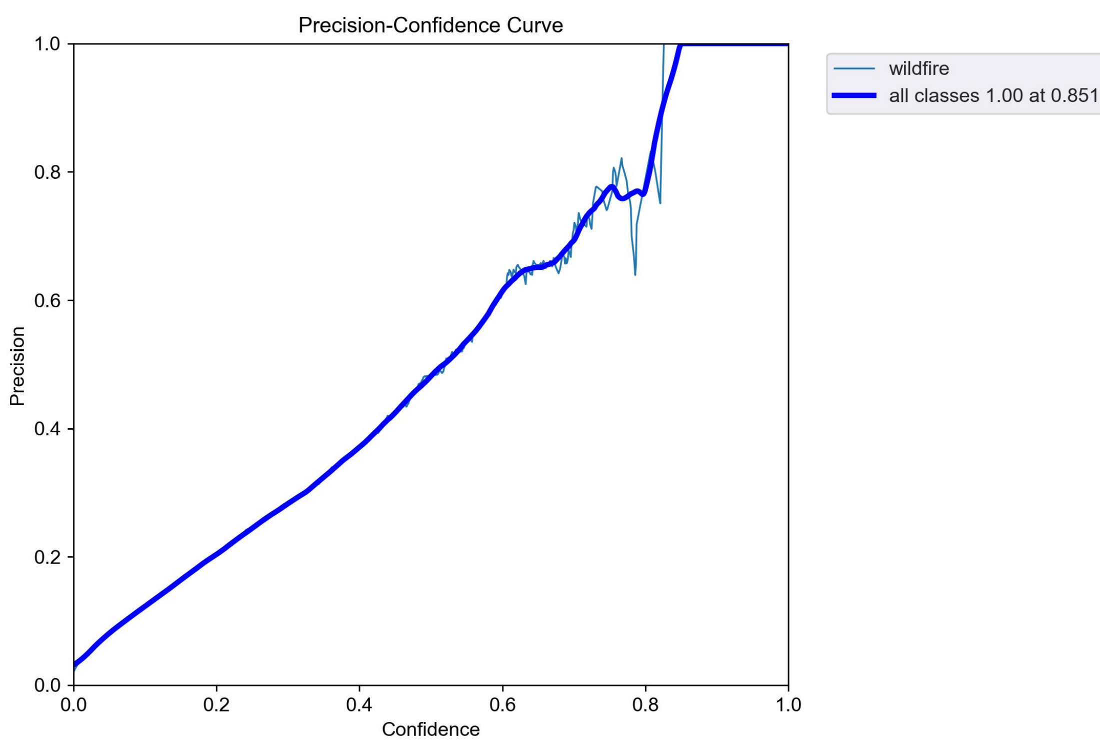

Fig. 8 Precision-Confidence Curve of YOLOv5s

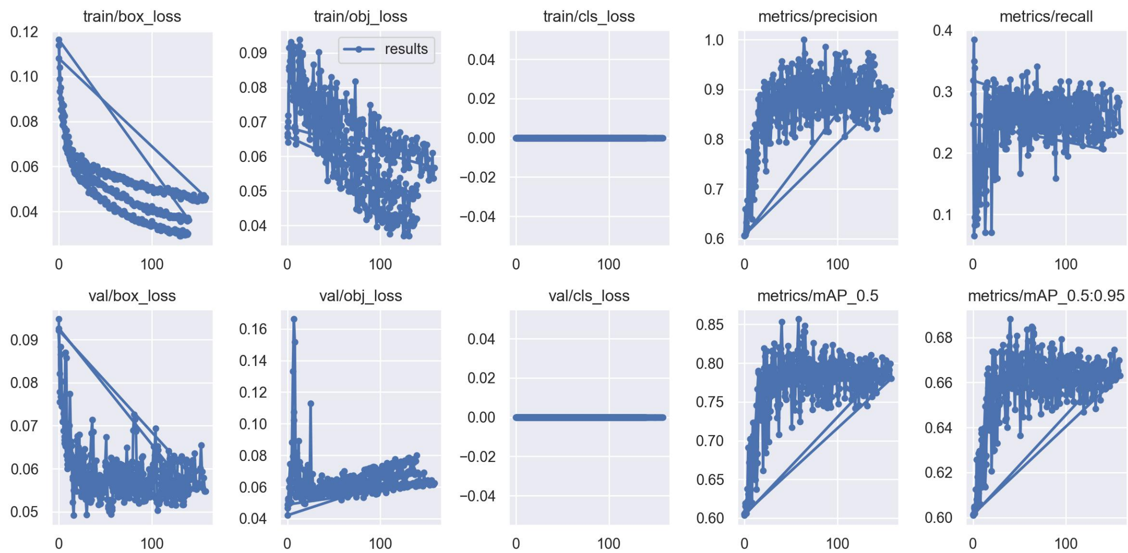

Fig. 9 Results of YOLOv5s

Fig. 10 Wildfire Prediction in validation of YOLOv5s

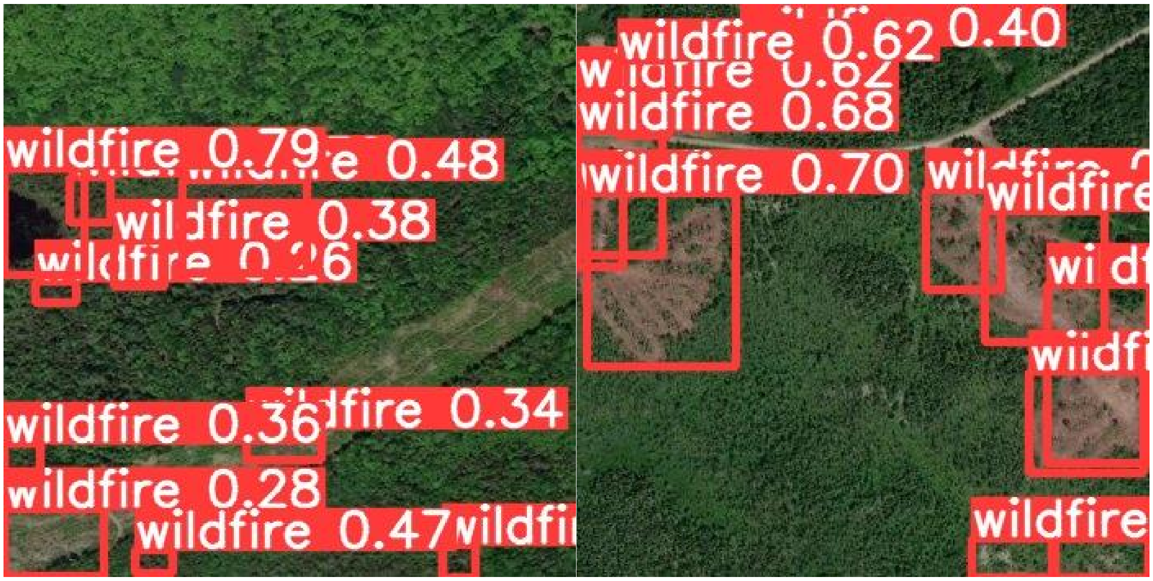

Fig. 11 Wildfire Detection in test of YOLOv5s

4.3.2 YOLOv5n

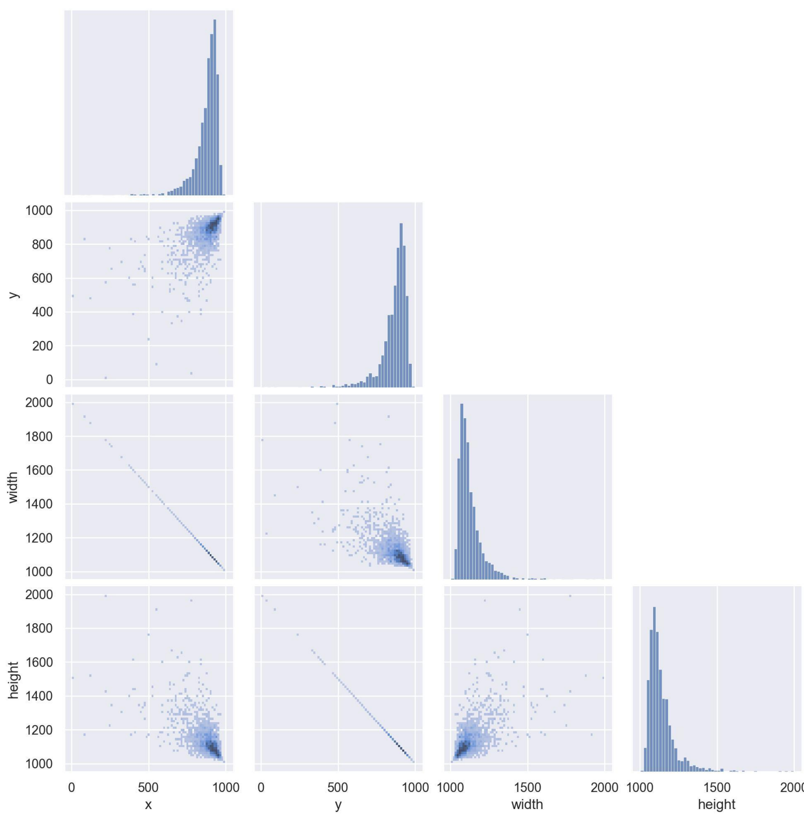

Fig. 12 The matrix diagram of the four variable joint distribution and correlation analysis of YOLOv5n

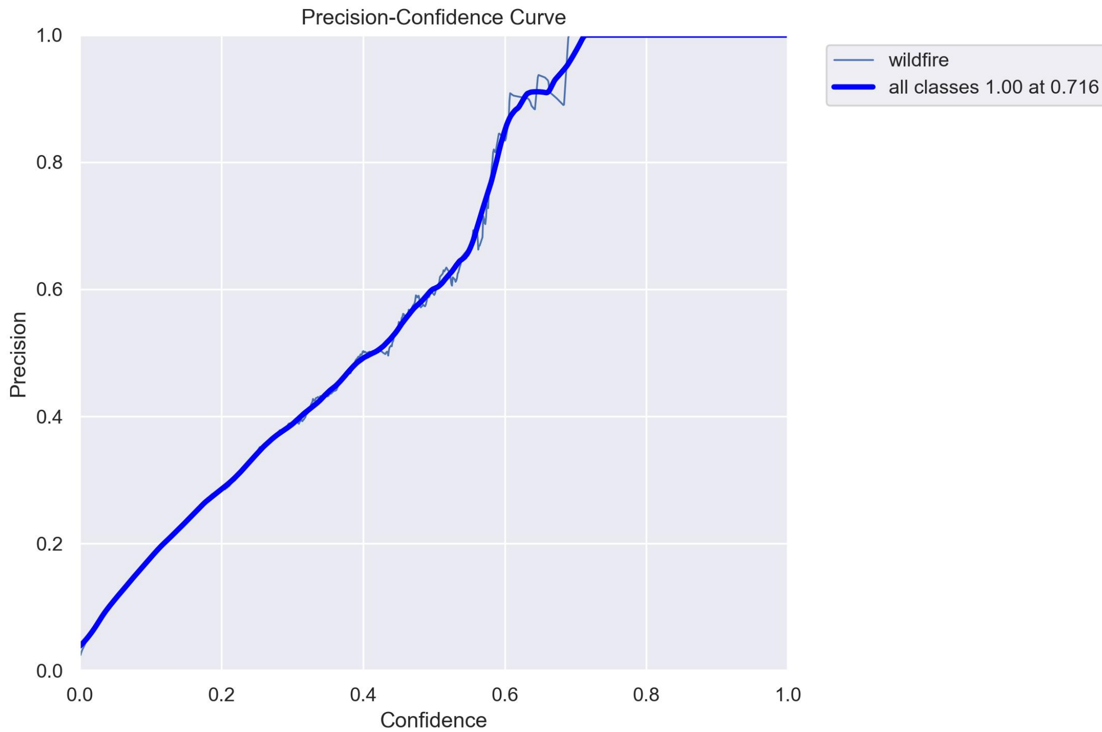

Fig. 13 Precision-Confidence Curve of YOLOv5n

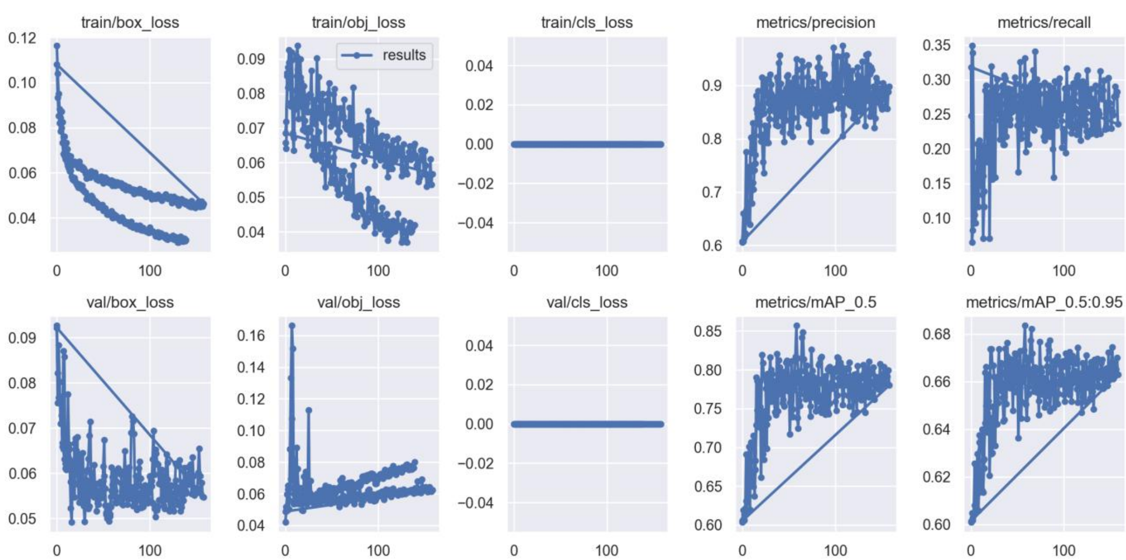

Fig. 14 Results of YOLOv5n

Fig. 15 Wildfire Prediction in validation of YOLOv5n

Fig. 16 Wildfire Detection in test of YOLOv5n

4.3.3 YOLOv5m

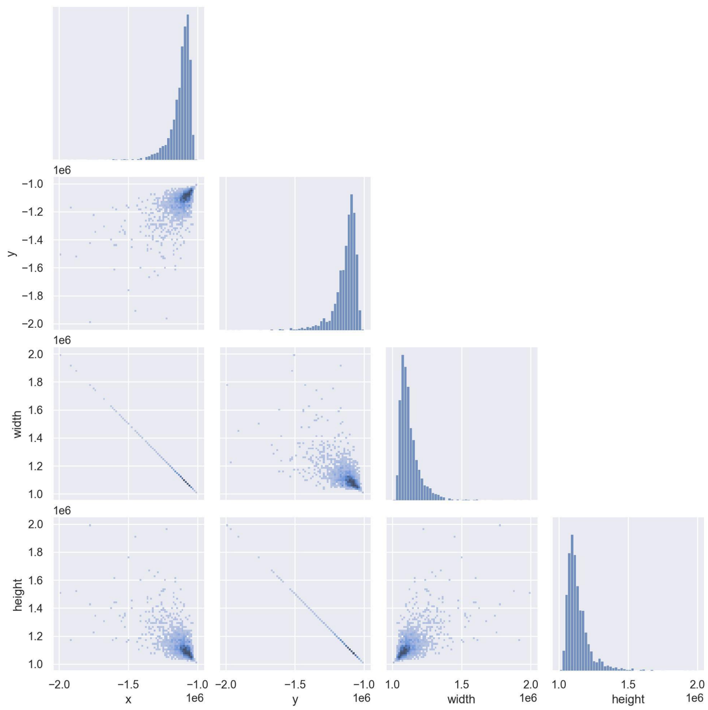

Fig. 17 The matrix diagram of the four variable joint distribution and correlation analysis of YOLOv5m

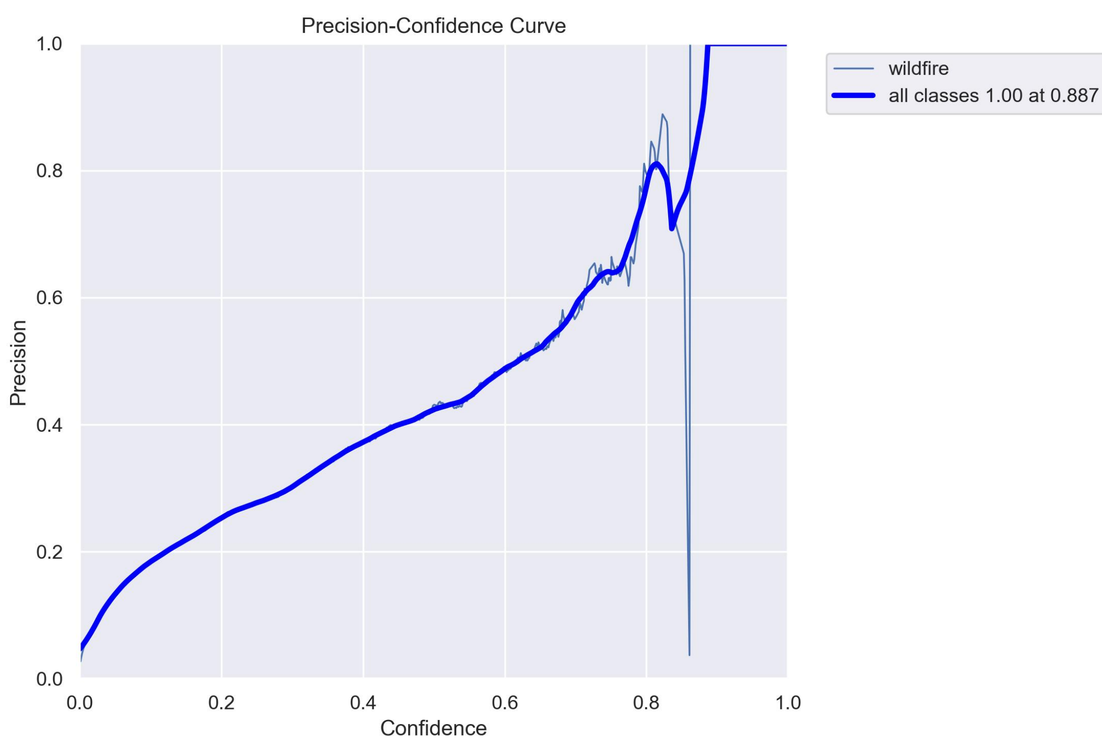

Fig. 18 Precision-Confidence Curve of YOLOv5m

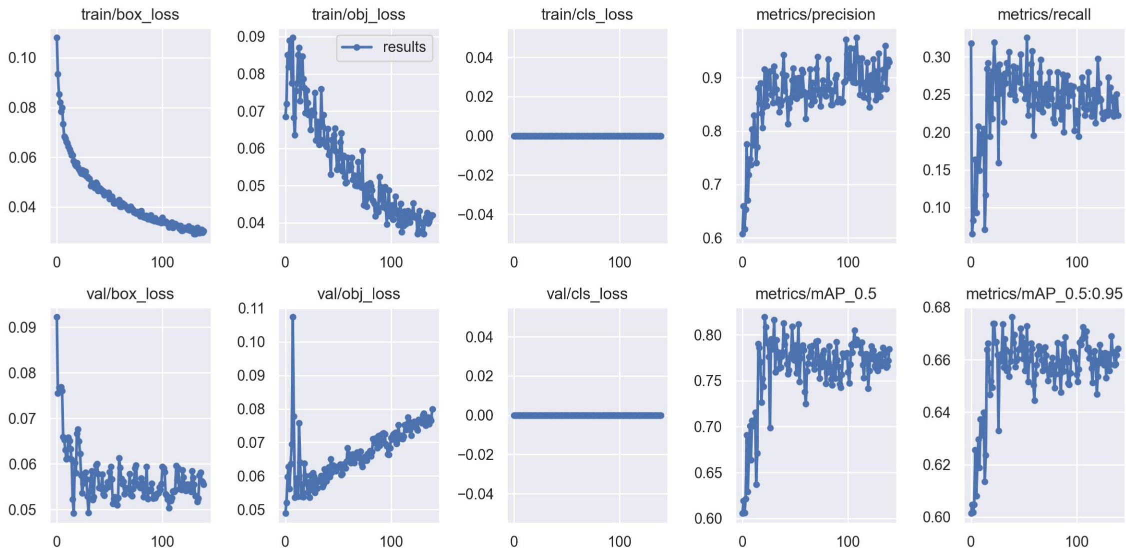

Fig. 19 Results of YOLOv5m

Fig. 20 Wildfire Prediction in validation of YOLOv5m

Fig. 21 Fig. 11 Wildfire Detection in test of YOLOv5m

4.3.4 YOLOv5l

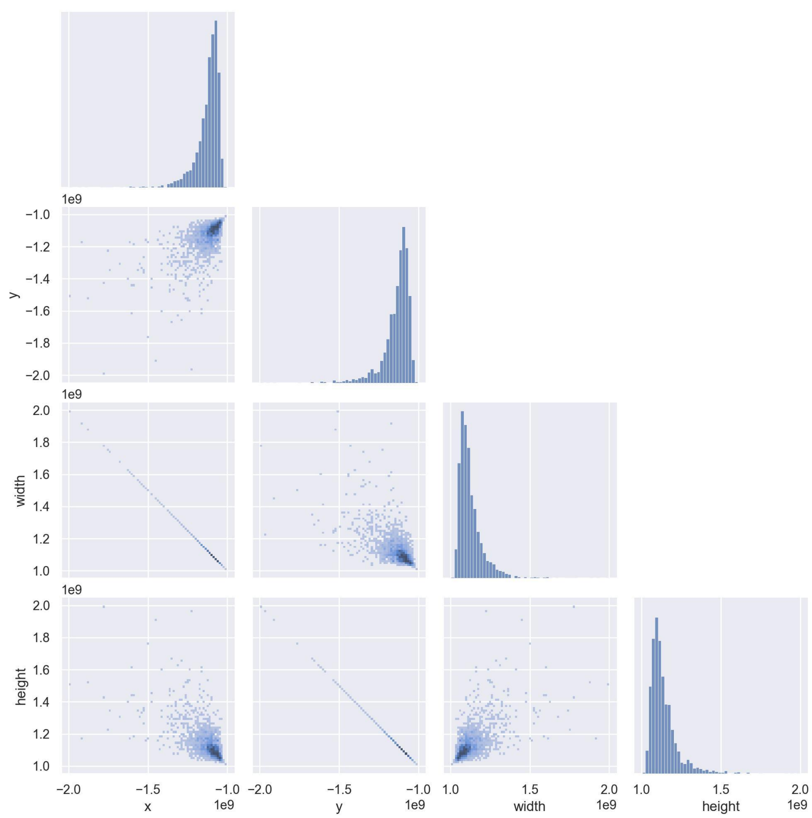

Fig. 22 The matrix diagram of the four variable joint distribution and correlation analysis of YOLOv5l

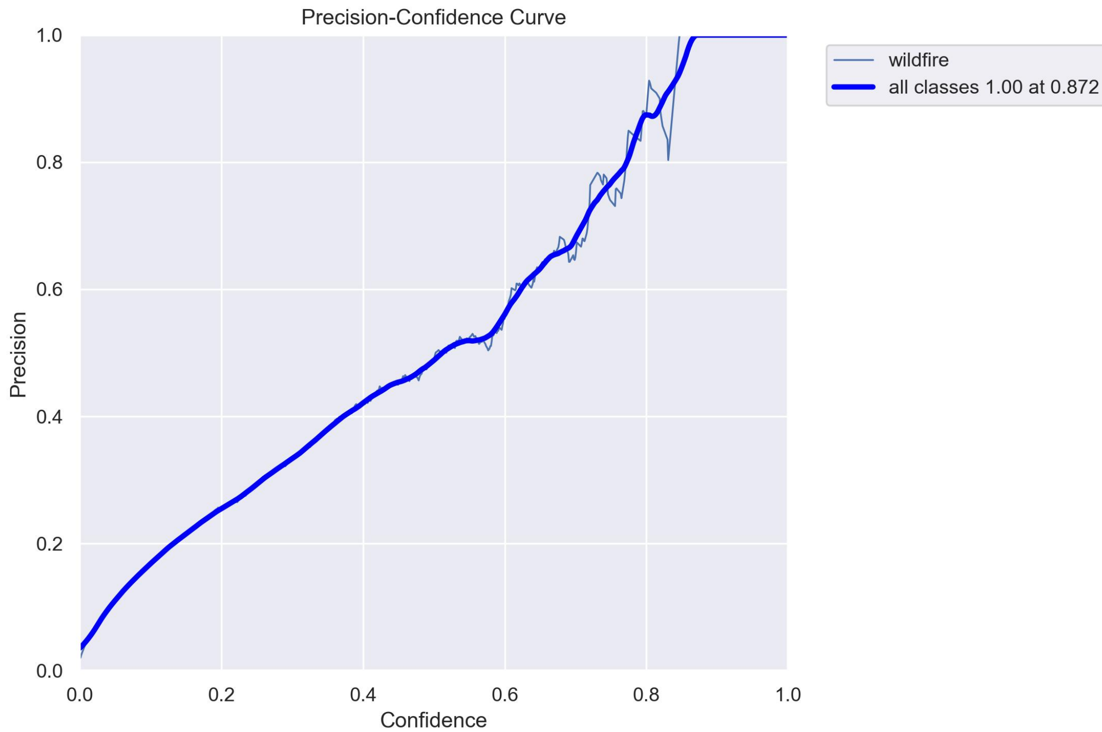

Fig. 23 Precision-Confidence Curve of YOLOv5l

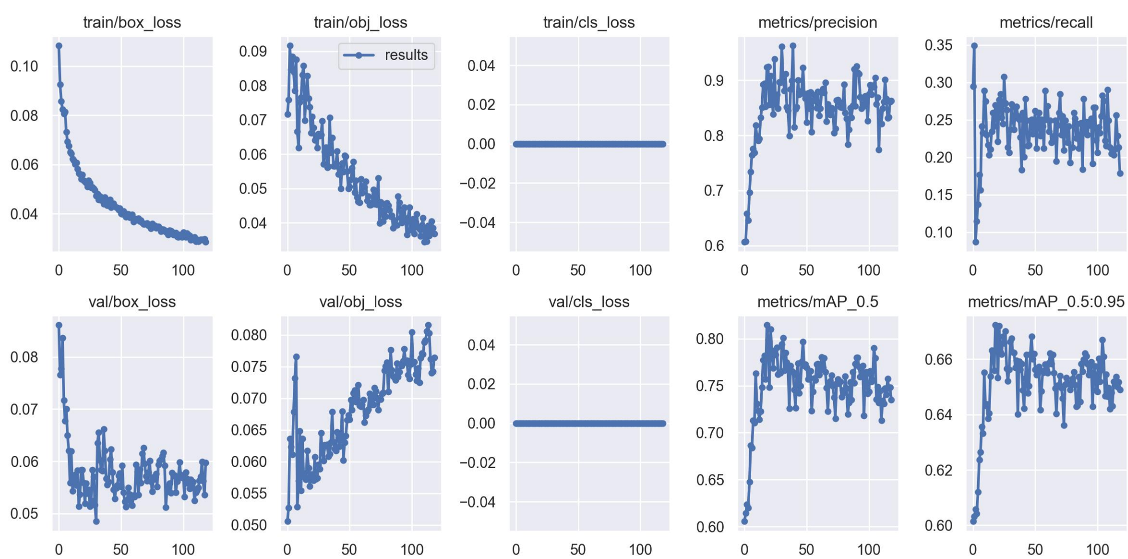

Fig. 24 Results of YOLOv5l

Fig. 25 Wildfire Prediction in validation of YOLOv5l

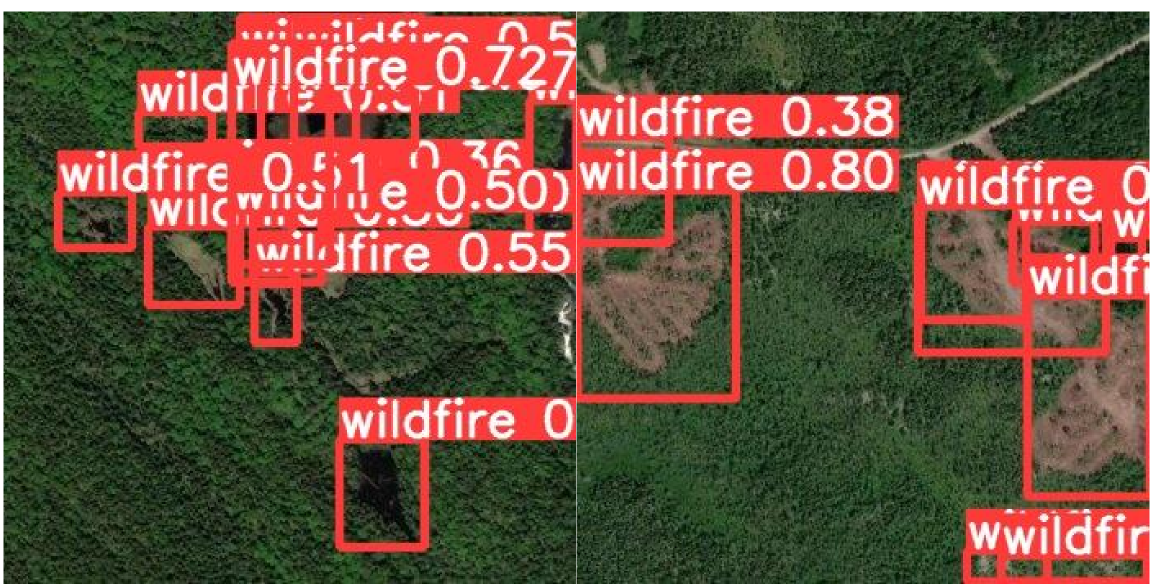

Fig. 26 Wildfire Detection in test of YOLOv5l

4.3.5 YOLOv5s-Fire

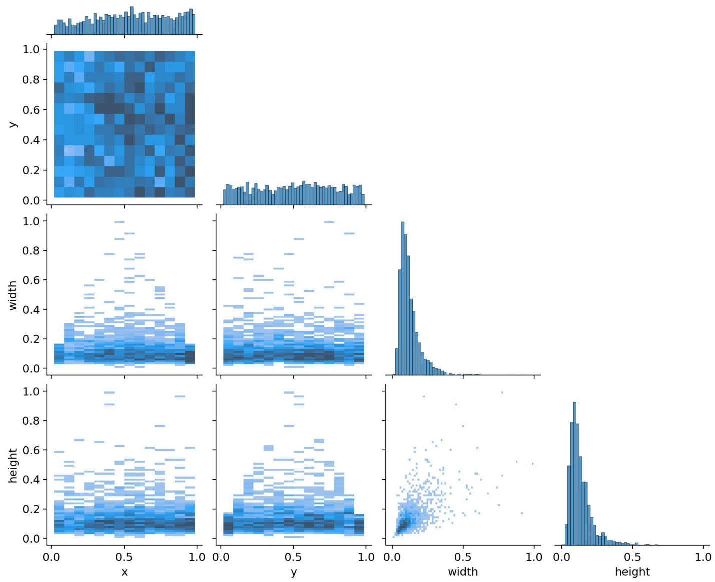

Fig. 27 The matrix diagram of the four variable joint distribution and correlation analysis of YOLOv5s-Fire

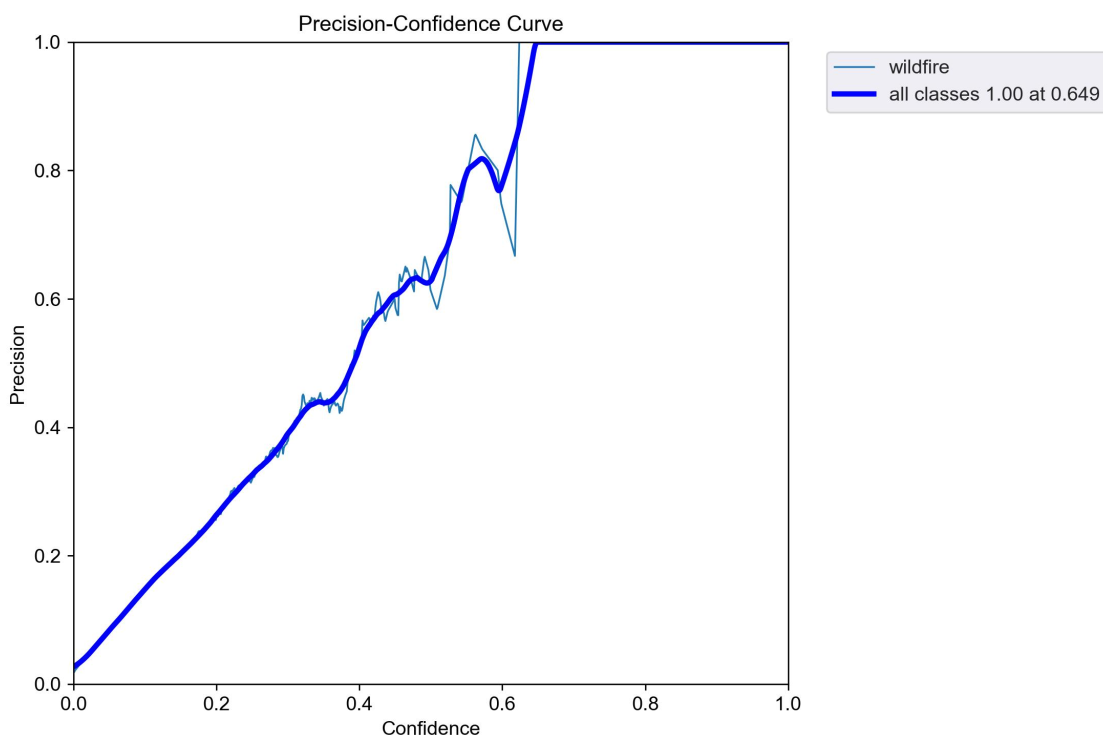

Fig. 28 Precision-Confidence Curve of YOLOv5s-Fire

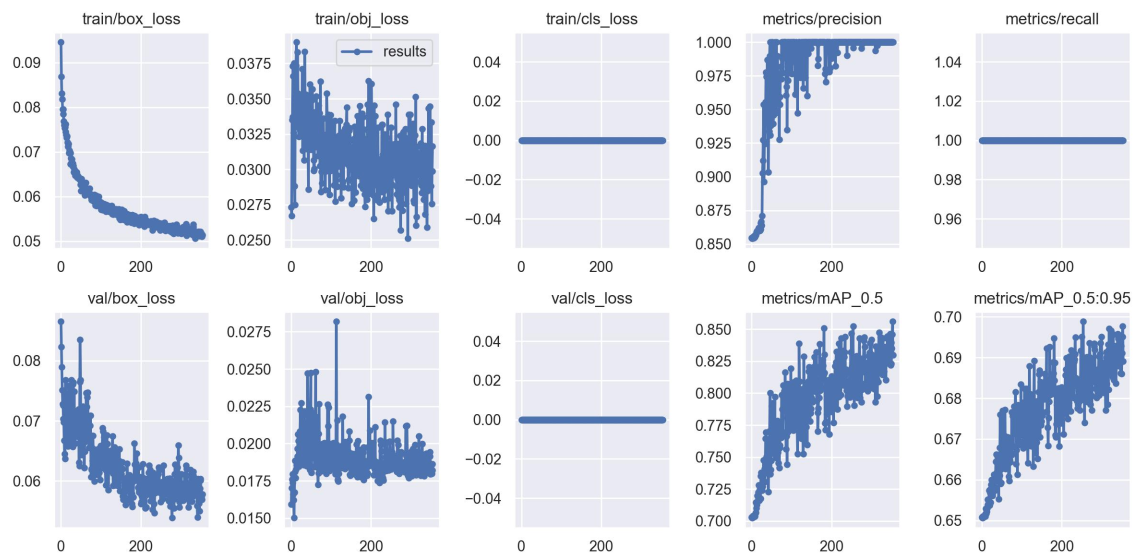

Fig. 29 Results of YOLOv5s-Fire

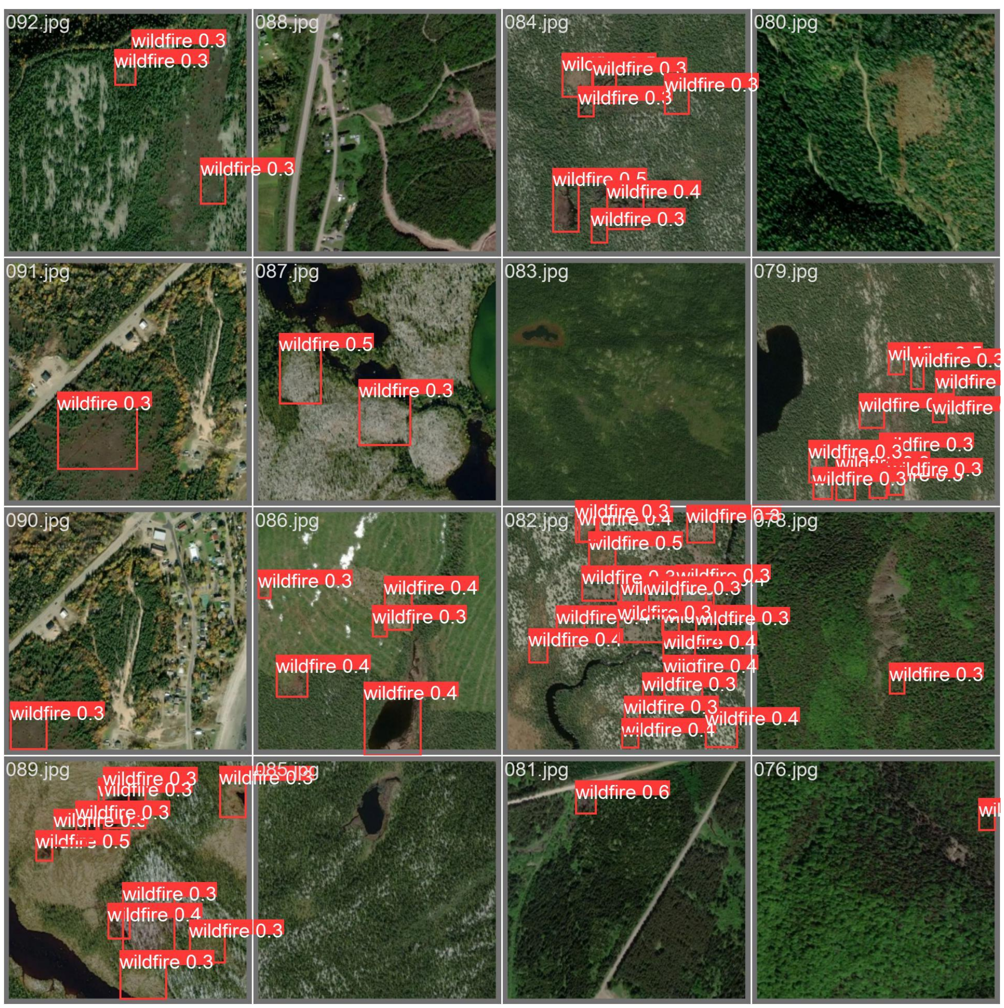

Fig. 30 Wildfire Prediction in validation of YOLOv5s-Fire

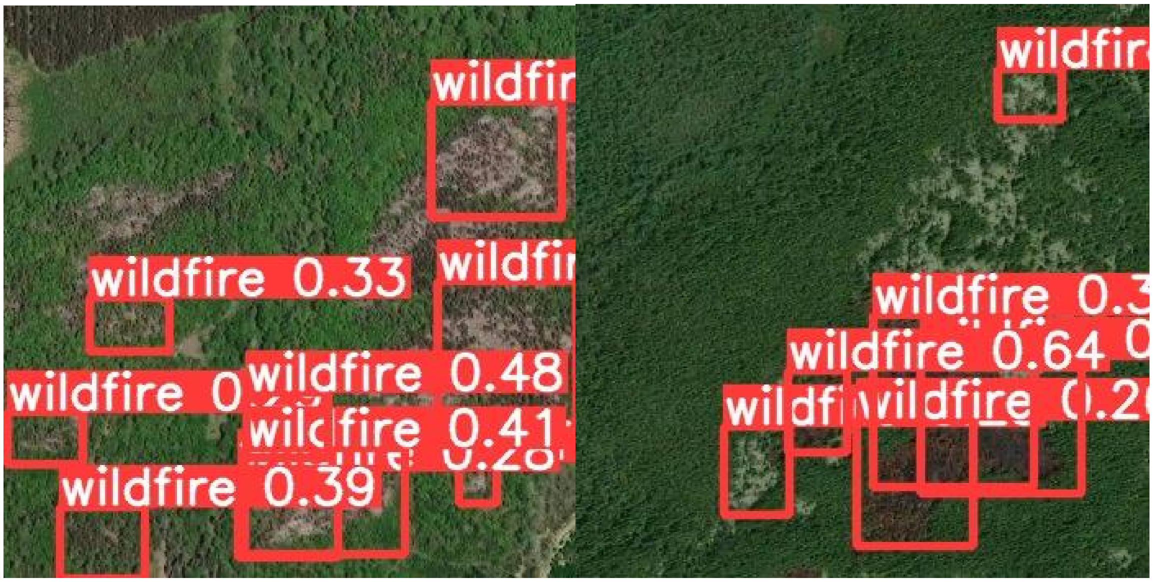

Fig. 26 Wildfire Detection in test of YOLOv5s-Fire

4.3.6 Comparison of models

The evaluation results comparison of models are shown in **Table 9**.

Model \ Metrics(%)	$Precision$	$Recall$	$mAP_{0.5}$	$mAP_{0.5:0.95}$	box_loss	obj_loss
YOLOv5s	94.8	91.8	85.3	68.86	3.62	4.86
YOLOv5n	94.9	93	85.8	68.41	4.63	5.67
YOLOv5m	94.3	85.6	81.4	67.62	3.03	4.19
YOLOv5l	92.3	86.9	81.3	67.19	2.85	3.68
Proposed Model	96.7	94.1	87.5	73.1	2.31	2.45

Table 9 Comparison of different models' evaluation results

As shown in **Table 9**, the proposed improved model exhibits overall advantages in the object detection task. In terms of accuracy, the model refrests the upper limit of performance with an precision of 96.7%, which is 1.8% higher than the suboptimal model YOLOv5n. At the same time, the recall rate of 94.1% is maintained, which is 2.3% optimized compared with the baseline model YOLOv5s, reflecting a better false detection and miss detection balance ability. Its $mAP_{0.5}$ index reaches 87.5%, which is 1.7% higher than that of its competitor YOLOv5n. In particular, its $mAP_{0.5:0.95}$ index that measures the adaptability of complex scenes, significantly outperforms all comparison models with 73.1%, and is 4.24% higher than that of YOLOv5s model. The robustness of the model to multi-scale objects is verified. In the training optimization, the model shows outstanding convergence characteristics: the target loss is 0.57% lower than that of lightweight YOLOv5n, and even better than that of YOLOv5l with more parameters. Among them, the positioning error box_loss also reaches the lowest level of the whole series, which is 0.19% lower than the original optimal model YOLOv5l. The overall data show that the improved model achieves more stable gradient propagation and feature localization capabilities through structural optimization. The comprehensive performance of the improved YOLOv5l model almost exceeds that of the larger scale YOLOv5L model while maintaining the lightweight architecture, which provides a cost-effective solution for the deployment of wildfire detection systems.

Chapter 5 Professional Issues

5.1 Project Management

5.1.1 Activities

Objective	Activities
1. Review the relevant literature and evaluate existing forest fire monitoring models.	➢ Research and read related papers and codes online to get ideas. ➢ Research and collect deep leaning algorithm for forest fire detection. ➢ Search for similar automatic fire detection model and make a comparison table. ➢ Complete the literature review.
2. Select and collect appropriate data	➢ Search and download suitable datasets.
3. Preprocess the image data.	➢ Convert image data to model acceptable formats, and tools such as LabelImg were used to label the fire area in the image
4. Split the data into training, validation, and test sets.	➢ Generate the training data set. ➢ The training set is used to learn the model parameters. ➢ Validation sets are used to adjust the hyperparameters and monitor performance of the models. ➢ Using the test set to evaluate the accuracy of the model.
5. Design an improved YOLOv5 model suitable for satellite wildfire identification.	➢ From the existing CNN models, YOLOv5 model is selected for improvement. ➢ Considering to improve the model performance of forest fire recognition. ➢ Optimized design of YOLOv5 algorithm.
6. Use the training set to train the constructed model.	➢ In the training process, the backpropagation algorithm is used to update the weights and biases in the network, so as to gradually learn the fire characteristics in the image.
7. Use verification sets to evaluate the performance of the trained model on new data.	➢ Evaluate the performance of the model using metrics such as accuracy, precision, recall rate, and adjust and improve it as needed.

8. Use test sets to test the accuracy of the model on the forest fire detection.	➢ Use the trained model test the accuracy of the location identification of active fires. ➢ Use the trained model test the precision of the estimate the burned area ➢ Use the trained model predicts new image, judging the severity of the fire.
9. Presenting the results of the research to the audience.	➢ Show the innovation of CNN model. ➢ Summarize its working process. ➢ Create PPT to present research findings.

Table 10 Activities table

5.1.2 Schedule

Below is the schedule for this project, it starts on October 21st 2024 and ends on April 1st 2025. The details are as follows:

- Literature review - 3 week
- Complete project proposal - 2 weeks
- Comparison of existing models - 3 weeks
- Database selection and download - 3 week
- Data preprocessing - 3 week
- Data set partition - 2 week
- Complete progress proposal - 4 week
- Build the model - 4 weeks
- Training model - 3 weeks
- Evaluation model - 3 weeks
- Test model - 3 weeks
- Complete the final report - 9 weeks
- Create PPT - 3 week

The schedule is represented by Gantt as below.

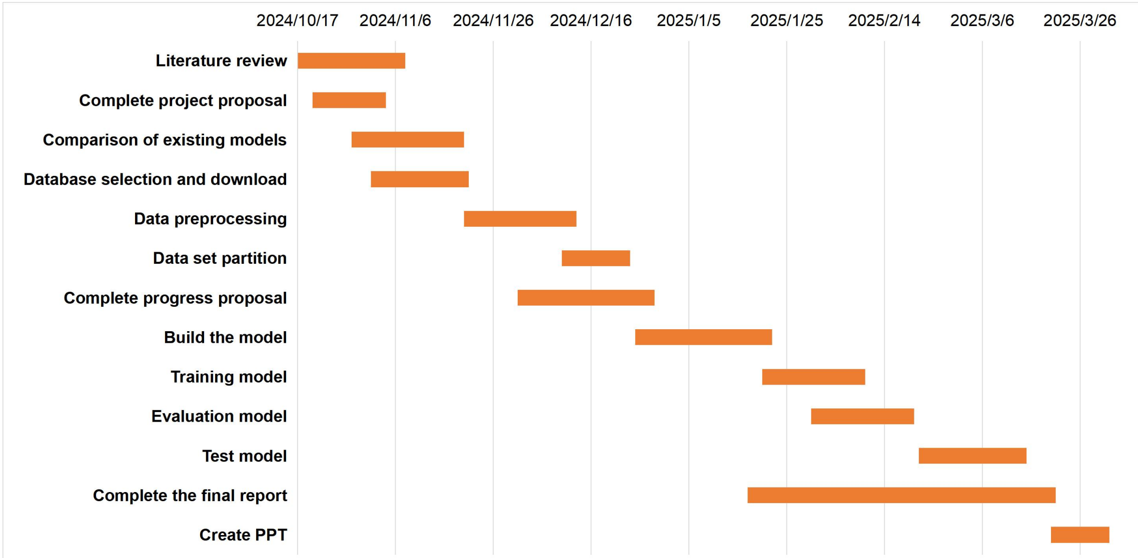

Fig. 31 Gantt chart for schedule

5.1.3 Project Data Management

a) Datasets was downloaded from Kaggle Wildfire Prediction Dataset (Satellite Images), links are as follows: https://www.kaggle.com/datasets/abdelghaniaaba/wildfire-prediction-dataset.

b) Relevant literature resources will be downloaded from academic websites such as Google Scholar and Semantic Scholar.

c) Using MyBib to manage references.

d) All documents and work will be stored on the flash drive and will be automatically backed up to the Baidu Cloud drive.

e) Baidu Cloud Link is: https://pan.baidu.com/s/1Q67PsmEjM_C2G1W-kq2p_Q?pwd=o52s. The structure of the file is shown below.

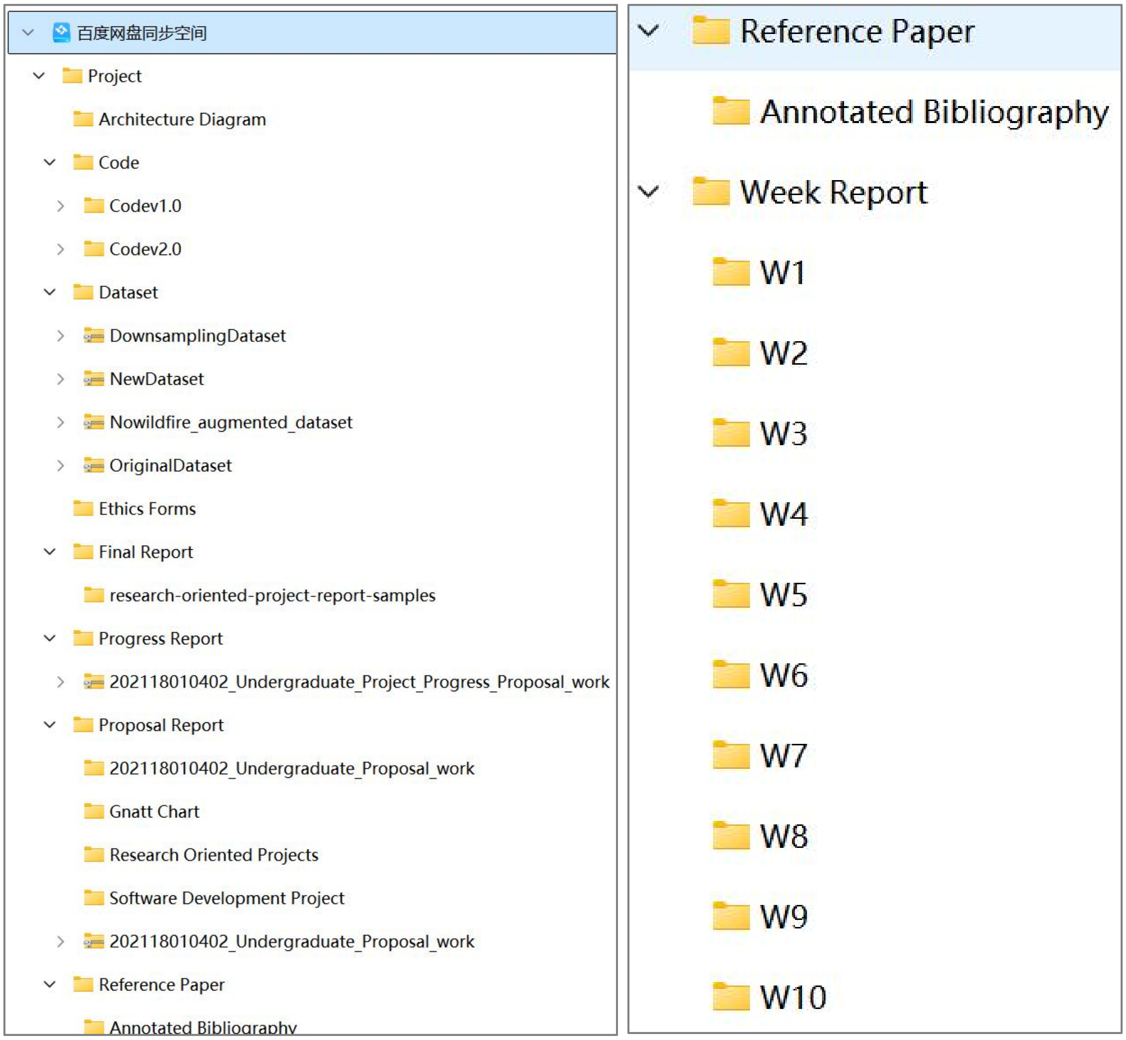

Fig. 32 Document structure

5.1.4 Project Deliverables

There are a total of 10 deliverables.

- Ethics form
- Project proposal
- Project progress report
- Weekly progress project reports
- Final project report
- Project code
- Project Diagram
- Project PPT
- Personal presentation
- Project dataset and website link

5.2 Risk Analysis

The risk analysis is shown in the figure below.

Risk ID	Potential Risk	Cause ID	Potential Causes	Severity	Likelihood	Risk	Mitigation ID	Mitigation
R1.1	Missed deadline	C1.1.1	Illness	1	2	2	M1.1.1	If ill, report special circumstances in advance.
		C1.1.2	Not sure the theme	2	1	2	M1.1.2	Meet with supervisor early to determine the research theme.
		C1.1.3	Poor time management	3	4	12	M1.1.3	Make the Gantt plan properly.
		C1.1.4	Unfinished project	5	3	15	M1.1.4	Submit weekly work report and report to supervisor.
R1.2	Feature creep	C.1.2.1	Over-ambitious project spec	3	2	6	M1.2.1	Discuss project plan with supervisor to create necessary and worthwhile goals.
R1.3	Error Model design	C1.3.1	Complexity of the Error Model	4	3	12	M1.3.1	Avoid overdesigning the model and design flexible error recovery strategies
		C1.3.2	Poor Test Coverage	3	2	6	M1.3.2	Increase test coverage related to data, model inference, and prediction processes, and ensure proper monitoring and logging.
R1.4	Loss of data	C1.4.1	Poor version control	5	3	15	M1.4.1	Implement two different version control strategies at the start of the project and regularly check the project versions.
R1.5	Risk of model overfitting	C1.5.1	Dataset Too Small	2	3	6	M1.5.1	Perform data augmentation on the dataset.
		C1.5.2	Insufficient Data Augmentation	3	3	9	M1.5.2	Increase the diversity of data augmentation strategies and adjust the intensity of augmentation.
		C1.5.3	Neglecting the Performance on the Test Set	4	2	8	M1.5.3	Use different evaluation metrics, cross-validation, and set up a monitoring metric for early stopping during training.
R1.6	Insufficient detection accuracy	C1.6.1	Unbanlanced Dataset	3	2	6	M1.6.1	Perform data augmentation to balance class distribution.
		C1.6.2	Improper model training parameters	4	3	12	M1.6.2	Tune the model hyperparameters, using an appropriate learning rate and loss function.
R1.7	Model inference is slow	C1.7.1	High model complexity	1	4	4	M1.7.1	Optimize the model structure, reduce the number of parameters and apply image downsampling
		C1.7.2	Insufficient Hardware performance	2	3	6	M1.7.2	Test the model on high-performance hardware.
R1.8	Data quality problem	C1.8.1	Low resolution of satellite images, cloud occlusion or noise	2	4	8	M1.8.1	Apply data preprocessing and combine other remote sensing data sources to improve data quality.
R1.9	Schedule delay	C1.9.1	Excessive time spent in the model optimization and evaluation phase	4	4	16	M1.9.1	Create a detailed timeline, prioritize completing core features before optimization.
R2.1	Project results do not meet expectations	C2.1.1	Performance metrics are not met	3	3	9	M2.1.1	Conduct multiple rounds of testing and evaluation in the mid and final stages of the projec.

Fig. 33 Risk Table

5.3 Professional Issues

5.3.1 Legal Issues

In the domain of forest fire detection, the satellite images employed might entail some sensitive geospatial data. Throughout this research, it is imperative to guarantee that the data utilized are fully safeguarded and the data collection process remains transparent, in alignment with the stipulations of the General Data Protection Regulation (GDPR) [24]. The YOLOv5 algorithm utilized in the fire detection model is protected under the Copyright, Designs and Patents Act of 1988 [25]. The satellite images employed in the project might possess specific licensing agreements to govern their usage, modification, and redistribution. Ensuring compliance with these licenses and respecting intellectual property rights is of paramount significance in evading legal disputes. Given that the model of this project can be utilized to guide decisions related to fire management and resource allocation, there might exist legal issues concerning liability. In the event that the system fails to accurately detect or assess the severity of the fire, thereby leading to injuries or delayed fire responses, there could be legal consequences regarding negligence or malfeasance. Developers are obligated to ensure the accuracy and reliability of their systems in order to mitigate legal risks.

5.3.2 Social Issues

The application of this model to forest fire detection alerts needs to ensure that it is available to a wide range of users, including government agencies, local authorities, and environmental organizations. The model should be designed to ensure that people with disabilities or with different levels of technical expertise can use it effectively. In addition, information about fire alerts should be provided to local communities, especially those in vulnerable areas. The use of satellite imagery and AI models in forest fire detection must be done carefully to avoid bias. If the training data of the YOLOv5 model is not diverse enough and it focuses mainly on forested areas in certain regions, then the model may not perform well in detecting fires in under-represented areas. There is a risk of inequity in the way fire alerts are generated in different geographical locations, which can lead to unequal allocation of resources and response efforts. In addition, the project could have a significant social impact, especially in communities that are regularly affected by forest fires. Timely and accurate fire detection can save lives and reduce the damage caused by fire. However, if a system is hastily deployed without proper consultation with the local community, it can lead to misunderstanding or lack of trust in the system, especially if there are false positives or false positives.

5.3.3 Ethical Issues

Ethical considerations for using machine learning in critical systems such as fire detection are crucial. According to the IEEE Code of Ethics [26], the development process must make clear how the system works, the limitations of the model, and any assumptions made during testing. Users should understand the decision-making process of the model and avoid blindly trusting the system, especially when the system affects critical decisions of fire management.

Given the significant life-or-death impact of forest fire detection, ensuring the accuracy and reliability of YOLOv5 models is a key ethical responsibility. Developers must strictly follow BCS guidelines [27] and conduct adequate testing to minimize errors and ensure that fire detection systems are not prone to false positives that cause unnecessary evacuations, or false negatives that cause delayed fire response.

In addition, this project involves satellite imagery and typically requires collaboration with agencies that control access to satellite data. Ethical concerns may be raised when considering the use of such data, especially when the purpose of the use of the data is not clearly disclosed to the public, which violates ACM guidelines [28]. Ensuring ethical use of satellite data also requires that data not be misused for commercial or political purposes without the consent of the communities being monitored.

5.3.4 Environmental Issues

a) Environmental Impact of Technology

The environmental impact of running AI models, especially deep learning algorithms like YOLOv5, is a growing concern. Training and deploying these models require substantial computational power, which leads to energy consumption. We must consider the environmental impact of the computational resources required for training the model, and explore methods to optimize energy consumption, such as leveraging energy-efficient hardware or cloud infrastructure powered by renewable energy.

b) Sustainability of Fire Management Systems

While the project may help in detecting and mitigating forest fires, it also indirectly raises questions about long-term sustainability. Relying on satellite imagery and AI may divert attention from other critical aspects of fire prevention, such as forest management, public education, and sustainable land use practices. There is an ethical responsibility to ensure that technology supports a broader environmental strategy rather than acting as a Band-Aid solution.

c) Accuracy in Environmental Monitoring

Inaccurate or incomplete detection of fires can exacerbate environmental damage. For example, undetected fires might spread further, increasing deforestation, air pollution, and biodiversity loss. Therefore, ensuring the environmental accuracy of the fire detection system is crucial to avoid exacerbating existing environmental issues.

Chapter 6 Conclusion

The paper proposes an enhanced remote sensing image forest fire detection model named YOLOv5s-Fire, which integrates multiple advanced modules. Based on the YOLOv5s framework, the model incorporates the CBAM attention mechanism, BiFPN feature fusion structure, GhostConv modules, and the SPP-Fire module that significantly improve detection accuracy and robustness while effectively controlling model size and computational cost. Through a series of experimental evaluations, the model architecture and hyperparameters were optimized to achieve superior classification performance. The results demonstrate that YOLOv5s-Fire outperforms all versions of the original YOLOv5 model across multiple evaluation metrics, achieving the 96.7% precision and 94.1% Recall. These results further confirm its efficiency and practicality in real-world remote sensing data processing, providing strong technical support for intelligent forest fire monitoring and emergency response. Future research will explore the integration of Transformer architectures to enhance the model's capability in capturing long-range dependencies and conduct generalization assessments and enhanced training on larger and more diverse datasets, facilitating the practical deployment of the proposed model in wildfire monitoring applications.

References

[1] S. T. Seydi, V. Saeidi, B. Kalantar, N. Ueda, and A. A. Halin, “Fire-Net: A Deep Learning Framework for Active Forest Fire Detection,” *Journal of Sensors*, vol. 2022, pp. 1–14, Feb. 2022, doi: https://doi.org/10.1155/2022/8044390.

[2] B. Özel, M. S. Alam, and M. U. Khan, “Review of Modern Forest Fire Detection Techniques: Innovations in Image Processing and Deep Learning,” *Information*, vol. 15, no. 9, Sep. 2024, doi: https://doi.org/10.3390/info15090538.

[3] A. Khan, B. Hassan, S. Khan, R. Ahmed, and A. Abuassba, “DeepFire: A Novel Dataset and Deep Transfer Learning Benchmark for Forest Fire Detection,” *Mobile Information Systems*, vol. 2022, pp. 1–14, Apr. 2022, doi: https://doi.org/10.1155/2022/5358359.

[4] S.-Y. Kim and A. Muminov, “Forest Fire Smoke Detection Based on Deep Learning Approaches and Unmanned Aerial Vehicle Images,” *Sensors*, vol. 23, no. 12, p. 5702, Jan. 2023, doi: https://doi.org/10.3390/s23125702.

[5] H. Yuan, Z. Lu, R. Zhang, J. Li, S. Wang, and J. Fan, “An effective graph embedded YOLOv5 model for forest fire detection,” *Computational Intelligence*, vol. 40, no. 2, Mar. 2024, doi: https://doi.org/10.1111/coin.12640.

[6] A. Bouguettaya, H. Zarzour, A. M. Taberkit, and A. Kechida, “A review on early wildfire detection from unmanned aerial vehicles using deep learning-based computer vision algorithms,” *Signal Processing*, vol. 190, p. 108309, Jan. 2022, doi: https://doi.org/10.1016/j.sigpro.2021.108309.

[7] H. Yar, T. Hussain, Z. A. Khan, D. Koundal, M. Y. Lee, and S. W. Baik, “Vision Sensor-Based Real-Time Fire Detection in Resource-Constrained IoT Environments,” *Computational Intelligence and Neuroscience*, vol. 2021, pp. 1–15, Dec. 2021, doi: https://doi.org/10.1155/2021/5195508.

[8] Saydirasulov Norkobil Saydirasulovich, Akmalbek Abdusalomov, Muhammad Kafeel Jamil, Rashid Nasimov, Dinara Kozhamzharova, and Young Im Cho, “A YOLOv6-Based Improved Fire Detection Approach for Smart City Environments,” *Sensors*, vol. 23, no. 6, pp. 3161–3161, Mar. 2023, doi: https://doi.org/10.3390/s23063161.

[9] M. Prakash, S. Neelakandan, M. Tamilselvi, S. Velmurugan, S. Baghavathi Priya, and Eric Ofori Martinson, “Deep Learning-Based Wildfire Image Detection and Classification Systems for Controlling Biomass,” *International journal of intelligent systems*, vol. 2023, pp. 1–18, Sep. 2023, doi: https://doi.org/10.1155/2023/7939516.

[10] E. A. Kadir, H. Irie, and S. L. Rosa, "Modeling of Wireless Sensor Networks for Detection Land and Forest Fire Hotspot," in *2019 International Conference on Electronics, Information, and Communication (ICEIC)*, Auckland, New Zealand, Jan. 2019, pp. 1–5. doi: https://doi.org/10.23919/ELINFOCOM.2019.8706364.
[11] V. Samborska and H. Ritchie, "Wildfires," *Our World in Data*, Apr. 2024, Accessed: Oct. 25, 2024. [Online]. Available: https://ourworldindata.org/wildfires.
[12] B. Laszlo, R. Agoston, and Q. Xu, "Conceptual Approach of Measuring the Professional and Economic Effectiveness of Drone Applications Supporting Forest fire Management," *Procedia Engineering*, vol. 211, pp. 8–17, Aug. 2018, doi: https://doi.org/10.1016/j.proeng.2017.12.132.
[13] T. J. Hawbaker, V. C. Radeloff, A. D. Syphard, Z. Zhu, and S. I. Stewart, "Detection rates of the MODIS active fire product in the United States," *Remote Sensing of Environment*, vol. 112, no. 5, pp. 2656–2664, May 2008, doi: https://doi.org/10.1016/j.rse.2007.12.008.
[14] C. O. Justice *et al.*, "The MODIS fire products," *Remote Sensing of Environment*, vol. 83, no. 1–2, pp. 244–262, Nov. 2002, doi: https://doi.org/10.1016/s0034-4257(02)00076-7.
[15] L. Alzubaidi *et al.*, "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *Journal of Big Data*, vol. 8, no. 53, Mar. 2021, doi: https://doi.org/10.1186/s40537-021-00444-8.
[16] S. Geetha, C. S. Abhishek, and C. S. Akshayanat, "Machine Vision Based Fire Detection Techniques: A Survey," *Fire Technology*, vol. 57, no. 2, pp. 591–623, Nov. 2020, doi: https://doi.org/10.1007/s10694-020-01064-z.
[17] R. Alkhatib, W. Sahwan, A. Alkhatieb, and B. Schütt, "A Brief Review of Machine Learning Algorithms in Forest Fires Science," *Applied sciences*, vol. 13, no. 14, pp. 8275–8275, Jul. 2023, doi: https://doi.org/10.3390/app13148275.
[18] R. Xu, H. Lin, K. Lu, L. Cao, and Y. Liu, "A Forest Fire Detection System Based on Ensemble Learning," *Forests*, vol. 12, no. 2, Feb. 2021, doi: https://doi.org/10.3390/f12020217.
[19] Y. Kang, E. Jang, J. Im, and C. Kwon, "A deep learning model using geostationary satellite data for forest fire detection with reduced detection latency," *GIScience & Remote Sensing*, vol. 59, no. 1, pp. 2019–2035, Nov. 2022, doi: https://doi.org/10.1080/15481603.2022.2143872.

[20] Y. Ding, M. Wang, Y. Fu, and Q. Wang, "Forest Smoke-Fire Net (FSF Net): A Wildfire Smoke Detection Model That Combines MODIS Remote Sensing Images with Regional Dynamic Brightness Temperature Thresholds," *Forests*, vol. 15, no. 5, pp. 839–839, May 2024, doi: https://doi.org/10.3390/f15050839.

[21] J. Yang, H. Jiang, S. Wang, and X. Ma, "A Multi-Scale Deep Learning Algorithm for Enhanced Forest Fire Danger Prediction Using Remote Sensing Images," *Forests*, vol. 15, no. 9, pp. 1581–1581, Sep. 2024, doi: https://doi.org/10.3390/f15091581.

[22] R. Khanam and M. Hussain, "What is YOLOv5: A deep look into the internal features of the popular object detector," *arXiv (Cornell University)*, Jul. 2024, doi: https://doi.org/10.48550/arxiv.2407.20892.

[23] S. K. Jaiswal and R. Agrawal, "A Comprehensive Review of YOLOv5: Advances in Real-Time Object Detection," *International Journal of Innovative Research in Computer Science and Technology*, vol. 12, no. 3, pp. 75–80, May 2024, doi: https://doi.org/10.55524/ijircst.2024.12.3.12.

[24] GDPR, "General data protection regulation (GDPR)," *General Data Protection Regulation (GDPR)*, May 25, 2018. https://gdpr-info.eu/ (accessed Dec. 22, 2024).

[25] Government of UK, "Copyright Acts and related laws," *GOV.UK*, 2023. https://www.gov.uk/government/publications/copyright-acts-and-related-laws (accessed Dec. 22, 2024).

[26] IEEE, "IEEE Code of Ethics," *ieee.org*, Jun. 2020. https://www.ieee.org/about/corporate/governance/p7-8.html (accessed Dec. 22, 2024).

[27] BCS, "British Computer Society," *Bcs.org*, 2018. https://www.bcs.org/ (accessed Dec. 22, 2024).

[28] ACM, "ACM code of ethics and professional conduct," *Association for Computing Machinery*, 2018. https://www.acm.org/code-of-ethics (accessed Dec. 22, 2024).

[29] G. Naidu, Tranos Zuva, and Elias Mmbongeni Sibanda, "A Review of Evaluation Metrics in Machine Learning Algorithms," *Lecture notes in networks and systems*, vol. 724, pp. 15–25, Jan. 2023, doi: https://doi.org/10.1007/978-3-031-35314-7_2.

[30] O. Rainio, J. Teuho, and R. Klén, "Evaluation metrics and statistical tests for machine learning," *Scientific Reports*, vol. 14, no. 1, pp. 1–14, Mar. 2024, doi: https://doi.org/10.1038/s41598-024-56706-x.

[31] H. Dalianis, "Evaluation Metrics and Evaluation," *Clinical Text Mining*, pp. 45–53, 2018, doi: https://doi.org/10.1007/978-3-319-78503-5_6.

[32] Z. Zheng, P. Wang, W. Liu, J. Li, D. Ye, and Y. Ren, "Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression," in *Proc. AAAI*, vol. 34, no. 07, pp. 12993–13000, Apr. 2020. doi: 10.1609/aaai.v34i07.6999.
[33] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 779–788. doi: 10.1109/CVPR.2016.91.
[34] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Jun. 2010. doi: 10.1007/s11263-009-0275-4.
[35] T.-Y. Lin *et al.*, "Microsoft COCO: Common Objects in Context," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 740–755. doi: 10.1007/978-3-319-10602-1_48.