

UNDERGRADUATE PROJECT PROGRESS PROPOSAL

|  |  |
| --- | --- |
| **Project Title:** | **Forest Fire Detection and Severity Assessment Using Improved YOLOv5 on Satellite Imagery** |
| **Surname:** | **Li** |
| **First Name:** | **Qinglian (Jane)** |
| **Student Number:** | **202118010402** |
| **Supervisor Name:** | **IRFAN ULLAH** |
| **Module Code:** | **CHC 6096** |
| **Module Name:** | **Project** |
| **Date Submitted:** | **December 26th 2024** |

**Table of Contents**

[1 Introduction 3](#_Toc3772)

[1.1 Background 3](#_Toc23433)

[1.2 Aim 4](#_Toc23172)

[1.3 Objectives 4](#_Toc16994)

[1.4 Project Overview 4](#_Toc23948)

[1.4.1 Scope 4](#_Toc20527)

[1.4.2 Audience 4](#_Toc13968)

[2 Background Review 5](#_Toc26759)

[3 Technical Progress 6](#_Toc898)

[3.1 Approach 6](#_Toc16220)

[3.1.1 YOLOv5 6](#_Toc18200)

[3.1.2 DataSet 8](#_Toc6390)

[3.2 Technology 9](#_Toc22359)

[3.3 Testing and Evaluation Plan 10](#_Toc29366)

[3.3.1 Dataset Testing Plan 10](#_Toc1473)

[3.3.2 Model Testing Plan 11](#_Toc8983)

[3.3.3 Model Evaluation Strategy 11](#_Toc30180)

[3.3.4 Pipeline Testing Plan 13](#_Toc12929)

[3.4 Design and Implementation 13](#_Toc32079)

[3.4.1 Data Preprocessing 13](#_Toc31722)

[3.4.2 Model Construction and Optimization 15](#_Toc55)

[4 Project Management 16](#_Toc23394)

[4.1 Activities 16](#_Toc6236)

[4.2 Schedule 17](#_Toc8686)

[4.3 Project Version management 18](#_Toc32742)

[4.4 Project Data management 19](#_Toc360)

[4.5 Project Deliverables 20](#_Toc18512)

[5 Professional Issues and Risk 21](#_Toc30785)

[5.1 Risk Analysis 21](#_Toc26019)

[5.2 Professional Issues 21](#_Toc17484)

[5.2.1 Legal Issues 21](#_Toc6285)

[5.2.2 Social Issues 22](#_Toc23816)

[5.2.3 Ethical Issues 22](#_Toc22033)

[5.2.4 Environmental Issues 23](#_Toc14584)

[6 References 24](#_Toc490)

# **Introduction**

## **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, review the literature and provide a comparative analysis of existing approaches. Section 3 introduces the techniques required for the project research and the processing of data sets. The project management plan is presented in section 4. Section 5 presents the issues and risks of the professional and the reference paper for the proposal is demonstrated in section 6.

## **Aim**

The aim of this proposed project is to develop a deep learning framework based on YOLOv5 to quickly identify the location of active fires, estimate the burned area, and classify the severity of fires by analyzing satellite imagery.

## **Objectives**

There are nine objectives proposed of the project.

* Review the relevant literature and evaluate existing forest fire monitoring models.
* Select and collect appropriate data
* Preprocess the image data by re-sizing, cropping, and normalization.
* Split the data into training, validation, and test sets.
* Design an improved YOLOv5 model suitable for forest fire identification.
* Use the training set to train the constructed model.
* Use verification sets to evaluate the performance of the trained model on new data.
* Use test sets to test the accuracy of the model on the forest fire detection.
* Presenting the results of the research to the audience.

## **Project Overview**

### **Scope**

The purpose of this proposed project is to develop a deep learning framework based on YOLOv5 and assessment of satellite image data captured via MODIS, which aims to facilitate rapid detection, estimate burn areas and assessment of the severity of forest fires. 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.

### **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.

# **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](https://onlinelibrary.wiley.com/authored-by/Khan/Ali) 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

# **Technical Progress**

## **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.

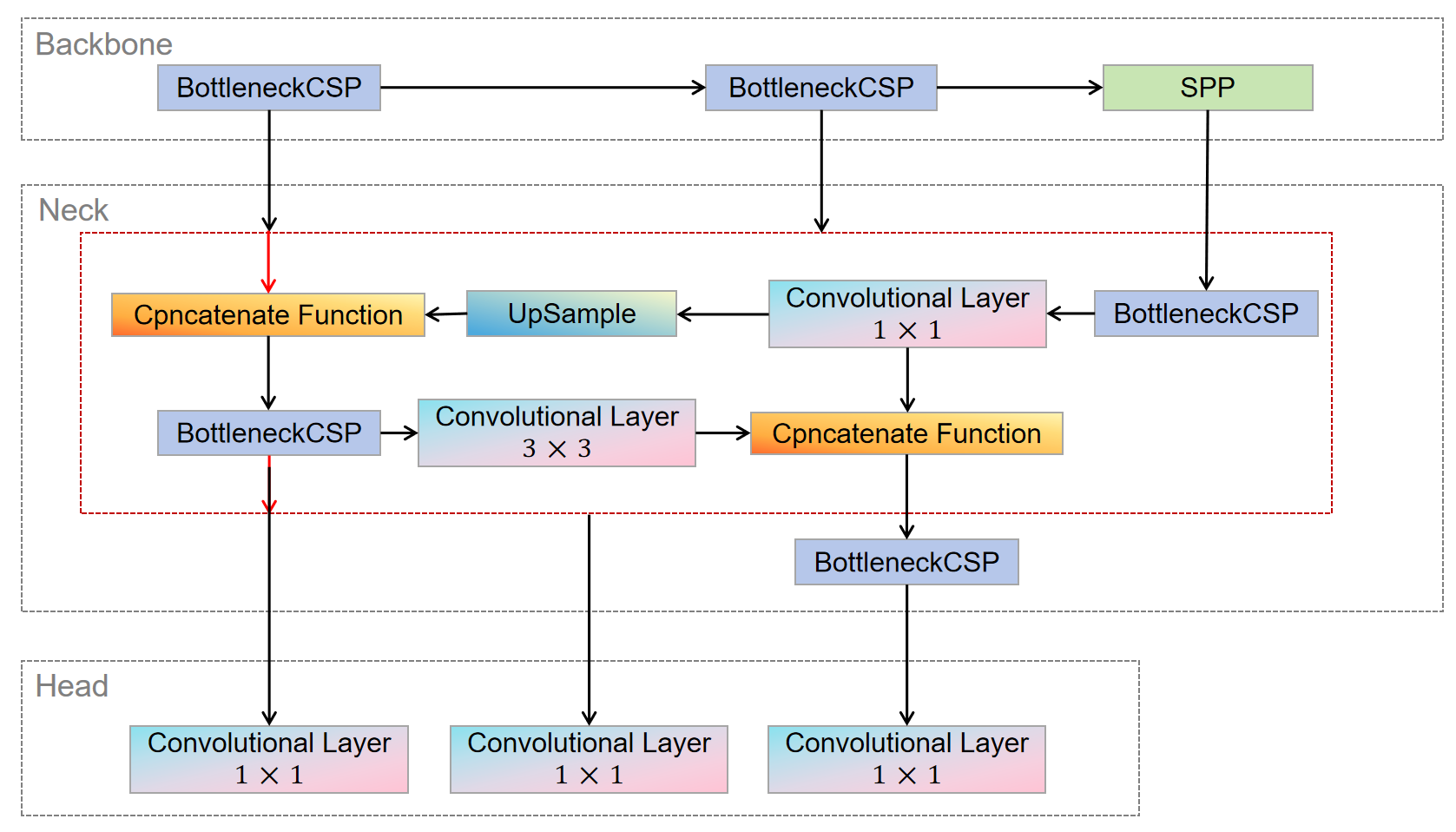


Fig. 1 YOLOv5 Structure

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 () 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.

|  |  |
| --- | --- |
|  | (1) |

**3.1.2 DataSet**

This proposed project intends to use the Wildfire Prediction Dataset (Satellite Images) from Kaggle for the classification problem of forest fires and no-fire images.

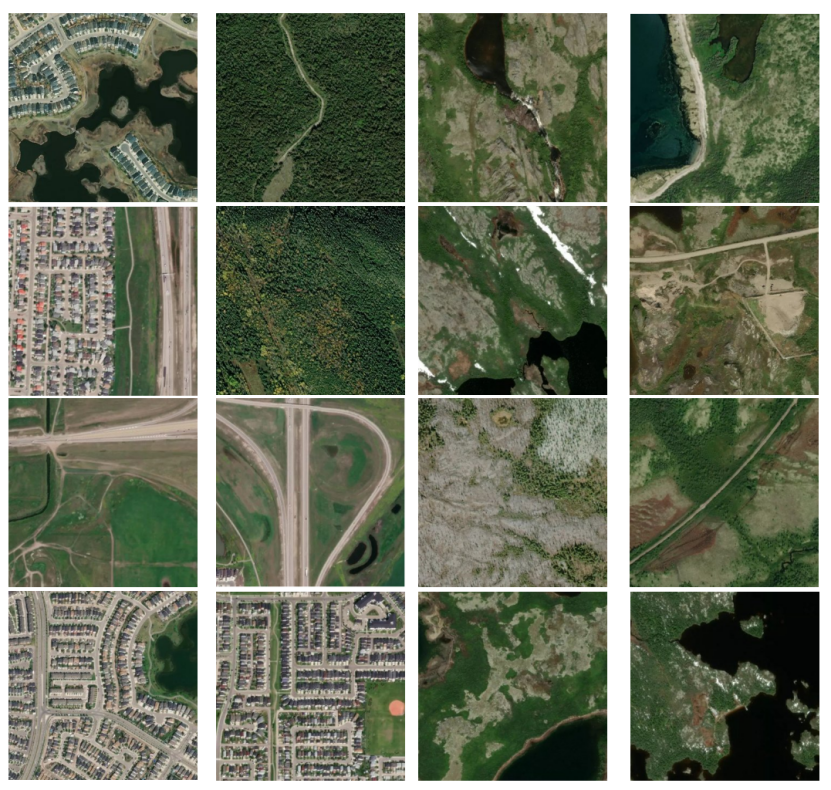


Fig. 2 Images of fire and no-fire classes

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**.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DataSet** | **Training** | **Testing** | **Validation** | **Total** |
| wildfire | 15898 | 3406 | 3406 | 22710 |
| No-wildfire | 14098 | 3021 | 3021 | 20140 |
| Total | 29996 | 6427 | 6427 | 42850 |

Table 2 Dataset splitting

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. All images have an initial resolution of 350 × 350 pixels. This project considers dividing the dataset into three classes as shown in **Table 2**, with 70% of the data used for training, 15% for testing, and the remaining 15% for validation.

## **Technology**

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

|  |  |  |
| --- | --- | --- |
| Software | Framework | TensorFlow2.0 |
| Language | Python 3.x |
| Operating System | Windows 11 |
| System development | IntelliJ IDEA 2023.2.1/ Visual Studio Code |
| Version management plan | Baidu Cloud |
| GitHub repository |
| Hardware | GPU | NVIDIA Tesla T4\*2/ NVIDIA Tesla P100 |
| CPU | The 11th Gen Intel® Core™ i5-11320H / i7-11390H Processors / CPU 4 cores |
| Memory | 16GB |
| SSD | 512B |

Table 3 Tools and techniques for development

The models discussed in this study were implemented using Python in a Honor Windows 11 system with 16 GB of RAM. The system features an Gen Intel ® Core TMi 7-11390H processor with a base clock speed of 3.20 GHz and a maximum turbo frequency of 4.50 GHz, providing substantial processing power for the model training and evaluation processes. The system is equipped with an NVIDIA GeForce MX450 integrated graphics card, but the performance is general, so this study chose to run on the kaggle platform, using the GPU is NVIDIA Tesla T4\*2 and NVIDIA Tesla P100.

## **Testing and Evaluation Plan**

The process of testing and evaluation is shown in **Fig. 3**.

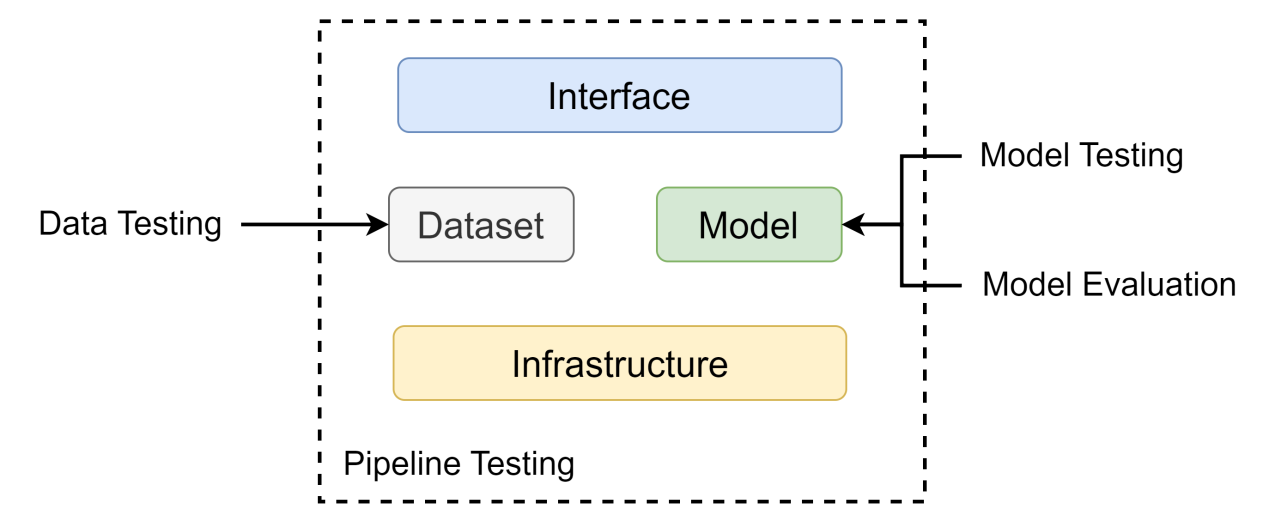


Fig. 3 Testing and Evaluation Process

**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. |
| 1. Check whether the resolution of the images is uniform, and adjust all the images to the model to the standard size required for training |
| 1. 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 |
| 1. 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. |
| 1. 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 4 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 Accuracy, Precision, Specificity, Recall, F1-value, Mean Intersection over Union (mIOU), Loss and Execution time. Each of the metrics is mathematically expressed as follows:

|  |  |
| --- | --- |
|  | (2) |

Accuracy is a metric to evaluate the overall classification performance of the model, and it indicates the proportion of all samples that the model predicts correctly [29], [31]. 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 [30], [31].

|  |  |
| --- | --- |
|  | (3) |

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.

|  |  |
| --- | --- |
|  | (4) |

Specificity is an indicator of the proportion of all negative samples predicted correctly over all actual negative samples [29].

|  |  |
| --- | --- |
|  | (5) |

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].

|  |  |
| --- | --- |
|  | (6) |

F1-score is a comprehensive metric to evaluate the classification performance of a model, which takes into account both accuracy and precision [29], [30], [31]. The higher the F1-score, the better the comprehensive performance of the model in the classification process, which can ensure high accuracy and precision at the same time.

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:

|  |  |
| --- | --- |
|  | (7) |

Here, is the number of classes, represents the true one-hot encoding, where for the correct class and for other classes. The is the predicted probability of class , typically output by the 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.

|  |  |
| --- | --- |
|  | (8) |

**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 5 Testing Plan of Pipeline

## **Design and Implementation**

**3.4.1 Data Preprocessing**

Data augmentation techniques play a vital role in the training of deep learning models, particularly when dealing with small-scale datasets or issues of class imbalance. In the Wildfire Prediction Dataset (Satellite Images), there exists a notable quantitative imbalance between the two categories of “wildfire” and “No-wildfire”. In this project, random data augmentation methods were applied to the images of the “No-wildfire” category to make their quantity equal to that of the “wildfire” category, thereby achieving a balanced data distribution and enhancing the generalization ability of the model.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DataSet** | **Training** | **Testing** | **Validation** | **Total** |
| wildfire | 15898 | 3406 | 3406 | 22710 |
| No-wildfire | 15898 | 3406 | 3406 | 22710 |
| Total | 31796 | 6812 | 6812 | 45420 |

Table 6 Dataset splitting

The distribution of the augmented dataset is shown in **Table 6**. The augmented dataset contains a total of 45,420 images, with 22,710 images as wildfire examples and the remaining 22,710 images as non-wildfire examples. All images have an initial resolution of 350×350 pixels.

Table 7. Dataset Segmentation

As shown in **Table 7**, after data augmentation, the number of images in the “wildfire” and “No-wildfire”categories was balanced across all subsets. The dataset was divided into three parts, with 70% of the data used for training, 15% for testing, and the remaining 15% for validation. During the data augmentation process, a series of random transformation techniques were applied to images in the “No-wildfire” category. These techniques included random horizontal flipping, random rotations, color jittering, and random cropping with resizing. **Fig. 4** demonstrate some random samples of “No-wildfire” category after data augmentation.



Fig. 4 Sample images of pre-processed No-wildfire

By enhancing the data of the “No-wildfire” category, this project significantly improved the model’s recognition ability for this category. Especially in situations where the category distribution is severely uneven, it effectively reduced the risk of overfitting caused by data bias. Simultaneously, data augmentation not only provides more diversified samples but also enhances the model’s ability to capture image features and robustness in different scenarios.

**3.4.2 Model Construction and Optimization**

This project utilizes the YOLOv5 model to achieve efficient detection and classification of “wildfire” and “No-wildfire” targets. As a lightweight and high-performance object detection model, YOLOv5 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 YOLOv5 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 YOLOv5 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**: The CIoU (Complete Intersection over Union) loss function was adopted to enhance the accuracy of bounding box regression, and Focal Loss was introduced to effectively address the impact of class imbalance on model performance.

**d) Transfer Learning**: Pretrained weights were loaded as initial model parameters, leveraging feature representations learned from the ImageNet dataset to accelerate model convergence and improve classification performance.

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.

# **Project Management**

## **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 monitoring software and make a comparison table. * Complete the literature review. |
| 1. Select and collect appropriate data | * Search and download suitable datasets. |
| 1. Preprocess the image data by re-sizing, cropping, and normalization. | * Convert image data to model acceptable formats, and the normalized processing, so that the input to the neural network. |
| 1. Split the data into training, validation, and test sets. | * Tools such as LabelImg were used to label the fire area in the image and 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. * The test set is used to evaluate the accuracy of the model. |
| 1. Design an improved YOLOv5 model suitable for forest fire identification. | * From the existing CNN models, YOLOv5 model is selected for improvement. * Considering the integration of YOLOv5 with other CNN models to improve the performance of forest fire recognition. * Optimized design of YOLOv5 algorithm. |
| 1. 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. |
| 1. 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. |
| 1. 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. |
| 1. 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 8 Activities table

## **Schedule**

Below is the schedule for this project, which 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.

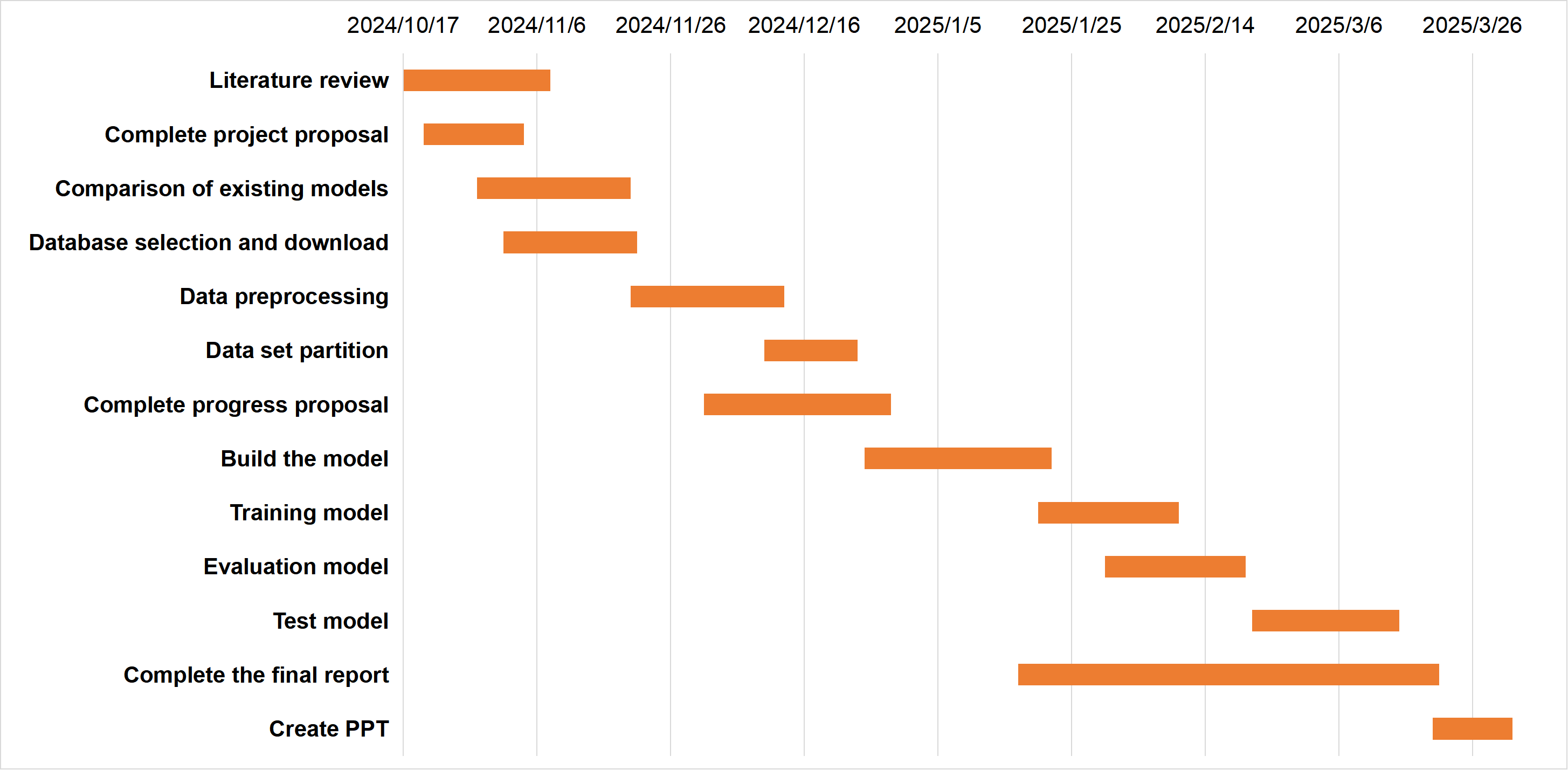


Fig. 5 Gantt chart for schedule

## **Project Version management**

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

|  |  |
| --- | --- |
| Version | Description |
| 1.0 | Implement YOLOv5 models |
| 2.0 | Implement the improved YOLOv5 model |
| 3.0 | Implement the integration of YOLOv5 with other CNN models |
| 4.0 | Optimized the integrated YOLOv5 model |

Table 9 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>.

## **Project Data management**

1. Datasets was downloaded from NASA FIRMS (the Fire Information for the Resource Management System), links are as follows: <https://firms.modaps.eosdis.nasa.gov/>.
2. Relevant literature resources will be downloaded from academic websites such as Google Scholar and Semantic Scholar.
3. Using MyBib to manage references.
4. All documents and work will be stored on the flash drive and will be automatically backed up to the Baidu Cloud drive.
5. Baidu Cloud Link is: <https://pan.baidu.com/s/1Q67PsmEjM_C2G1W-kq2p_Q?pwd=o52s>. The structure of the file is shown below.

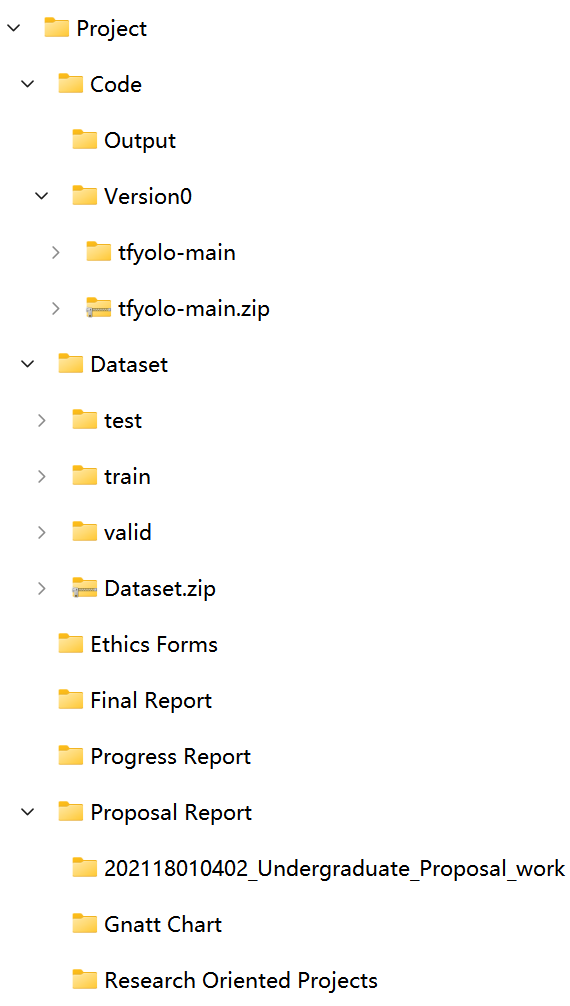
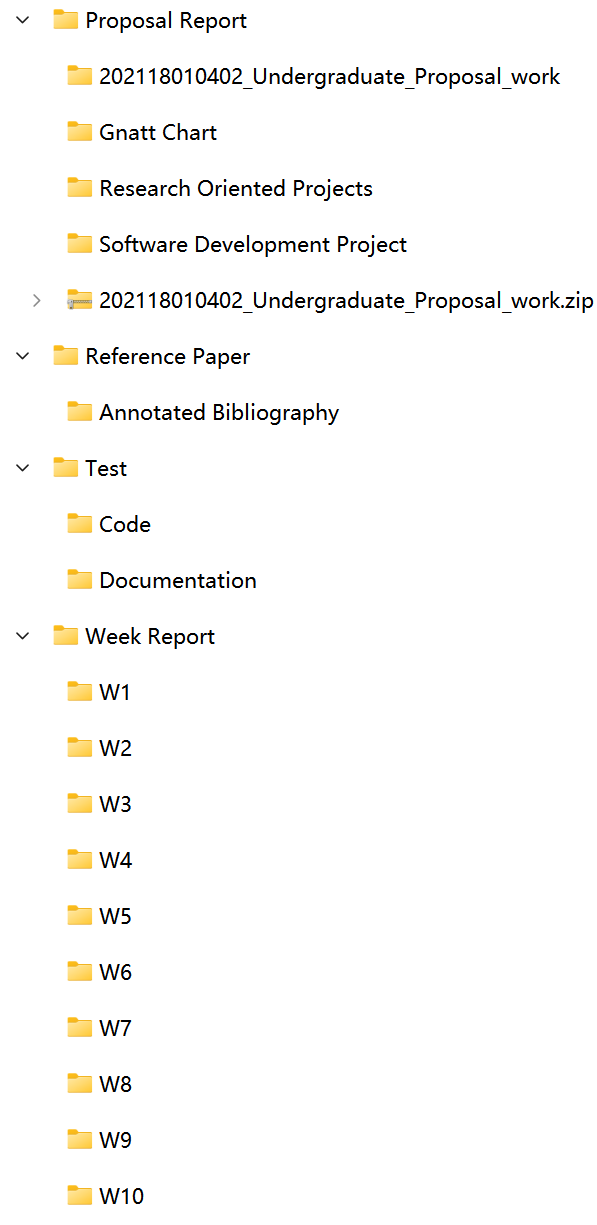
 

Fig. 6 Document structure

## **Project Deliverables**

There are a total of 9 deliverables.

* Ethics form
* Project proposal
* Weekly progress project reports
* Final project report
* Project code
* Project PPT
* Personal presentation
* Project dataset and website link
* Test results

# **Professional Issues and Risk**

## **Risk Analysis**

The risk analysis is shown in the figure below.

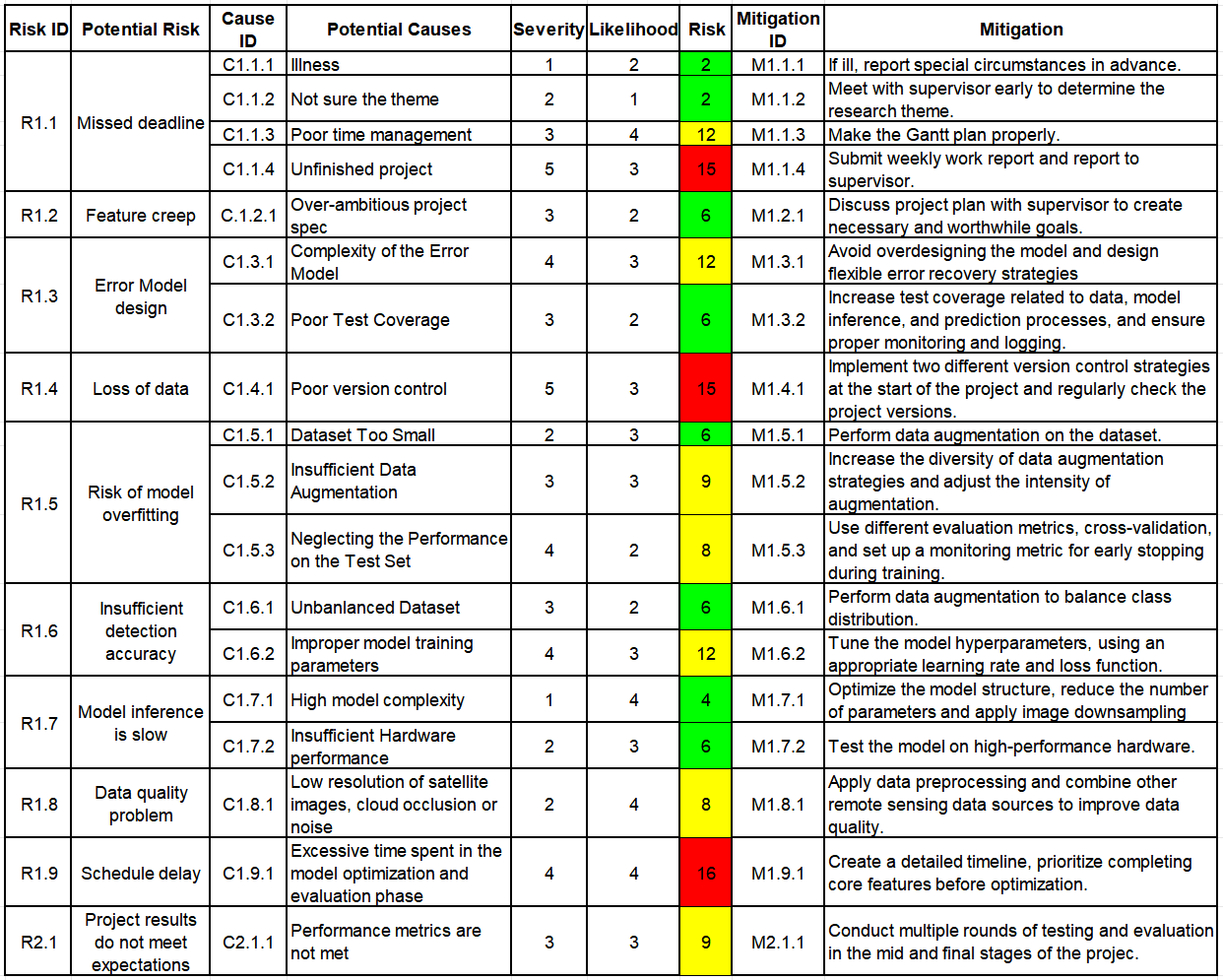


Fig. 7 Risk Table

## **Professional Issues**

**5.2.1 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.

**5.2.2 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.

**5.2.3 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.

**5.2.4 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.

# **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.](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.>