AIML Project Report Format

# INTRODUCTION

1.1Project Overview

Project Overview: YOLO-Based Safety Gear Surveillance System

Objective: The primary objective of this project is to develop a robust surveillance system utilizing the YOLO (You Only Look Once) object detection algorithm for real-time monitoring and enforcement of safety gear compliance in designated areas.

Components:

1.YOLO Object Detection:

Implement YOLO algorithm for accurate and efficient real-time object detection.

Train the model to recognize key safety gear items, including helmets, safety vests, goggles, etc.

2.Camera Integration:

Integrate surveillance cameras to capture live video feeds.

Ensure compatibility with various camera types and resolutions for versatile deployment.

3.Data Preprocessing:

Preprocess video feeds to optimize input for the YOLO model.

Address challenges such as noise, varying lighting conditions, and occlusions.

4.Safety Gear Database:

Maintain an annotated database of safety gear images for model training.

Regularly update the database to enhance the model's accuracy and adaptability.

Workflow:

1.Video Feed Processing:

Capture and process video feeds from connected surveillance cameras.

Employ YOLO for real-time object detection within the video streams.

2.Object Classification:

Classify detected objects into relevant safety gear categories.

Implement confidence score thresholds to filter out false positives.

3.Safety Gear Compliance Check:

Evaluate the presence and correct usage of safety gear on individuals in the video feed.

Implement logic to identify instances of missing or improperly worn gear.

4.Alert System:

Trigger alerts, such as visual cues or notifications, for non-compliance.

Ensure real-time alerting to designated personnel or safety officers.

User Interface:

Develop a user-friendly interface for system monitoring and management, featuring:

Live video feed viewing.

Historical data analysis.

Alert management functionalities.

Integration with Access Control: Optionally integrate the system with access control mechanisms to restrict entry for non-compliant individuals.

Reporting and Analytics:Generate reports on safety compliance trends.

Provide analytics for informed decision-making regarding safety protocols.

Security and Privacy:

Implement robust security measures to safeguard system integrity.

Ensure compliance with privacy regulations by securing and anonymizing captured data.

Testing and Evaluation:Conduct :

comprehensive testing to validate system accuracy and reliability.

Evaluate the system in real-world scenarios to identify and address potential challenges.

Maintenance and Updates:

Establish routine maintenance protocols for both hardware and software components.

Facilitate easy updates and model retraining for sustained effectiveness.

Deployment:

Deploy the system in targeted environments such as construction sites, factories, or industrial settings.

Provide on-site training for users and administrators.

Future Enhancements:

Explore AI-based behavior analysis for advanced safety monitoring.

Consider expanding the system for monitoring additional safety parameters.

This project aims to create an efficient and reliable YOLO-based safety gear surveillance system, contributing to improved safety standards and compliance in high-risk environments.

Top of Form

v

* 1. Purpose

1.Safety Compliance Assurance:

Ensure that individuals in monitored areas are wearing the required safety gear, such as helmets, safety vests, goggles, etc.

Minimize the risk of accidents and injuries by actively enforcing safety protocols through real-time surveillance.

2.YOLO Object Detection Implementation:

Utilize the YOLO (You Only Look Once) object detection algorithm for accurate and efficient real-time detection of safety gear.

Train the YOLO model to recognize and classify various safety gear items with a high level of precision.

3.Proactive Alert System:

Implement a proactive alert system to notify relevant personnel or safety officers immediately when safety gear compliance is compromised.

Ensure real-time alerts through visual cues, audible alarms, or notifications to prevent potential safety hazards.

4.User-Friendly Interface:

Develop an intuitive and user-friendly interface for easy monitoring and management of the surveillance system.

Provide features such as live video feed viewing, historical data analysis, and alert management to enhance user experience.

5.Integration with Access Control:

Optionally integrate the system with access control mechanisms to restrict entry for individuals not compliant with safety gear requirements.

Enhance overall security measures by combining safety surveillance with access control.

6.Data Reporting and Analytics:

Generate comprehensive reports on safety compliance trends over time.

Provide analytical insights to management for informed decision-making regarding safety protocols and resource allocation.

7.Privacy and Security Measures:

Implement robust security measures to protect the integrity of captured data.

Ensure compliance with privacy regulations by incorporating anonymization and encryption techniques.

8.Scalability and Adaptability:

Design the system to be scalable and adaptable to different environments and industries.

Allow for easy integration with existing infrastructure and future expansion to accommodate evolving safety standards.

9.Continuous Improvement:

Establish mechanisms for continuous improvement through regular updates and model retraining.

Stay abreast of technological advancements to incorporate the latest innovations in safety surveillance.

10.Facilitate Industry Compliance:

Support industries in meeting safety regulations and compliance standards.

Act as a proactive tool to prevent accidents, reduce liabilities, and enhance overall workplace safety.

The ConstructGuard project aims to make a significant contribution to workplace safety by deploying an advanced YOLO-based safety gear surveillance system. Through proactive monitoring and real-time alerts, the system strives to create a safer working environment and promote a culture of adherence to safety protocols within high-risk settings.

Top of Form

# LITERATURE SURVEY

* 1. Existing problem

1. Limited Dataset Variability:

Many YOLO-based safety gear surveillance systems face challenges due to a limited dataset for training. The variability in safety gear types, colors, and environmental conditions may not be adequately represented, leading to suboptimal model performance in real-world scenarios.

2. Occlusion Handling:

The issue of occlusion, where safety gear items are partially or fully obscured in the camera feed, poses a significant challenge. Existing systems may struggle to accurately detect and classify safety gear under such conditions, impacting the overall reliability of the surveillance system.

3. Real-Time Processing Efficiency:

Achieving real-time processing efficiency remains a concern. Some literature highlights the need for optimization techniques to ensure that the YOLO model can process high-resolution video feeds efficiently without compromising on accuracy, especially in environments with multiple concurrent activities.

4. False Positive and False Negative Rates:

Literature reports on challenges related to false positives and false negatives in safety gear detection. These issues can lead to unnecessary alerts or, conversely, the failure to detect individuals not compliant with safety regulations, reducing the overall effectiveness of the surveillance system.

5. Adaptability to Dynamic Environments:

The adaptability of YOLO-based systems to dynamic environments, where lighting conditions and the layout of the monitored area can change rapidly, remains an ongoing concern. Existing literature suggests the need for models that can adapt in real-time to such environmental variations.

6. Integration with Access Control Systems:

While there is potential for integrating safety gear surveillance systems with access control mechanisms, challenges exist in seamless integration. Literature points out the need for standardized protocols and interfaces to enable effective collaboration between safety surveillance and access control systems.

7. Ethical and Privacy Considerations:

Ethical and privacy concerns related to constant surveillance are raised in the literature. Striking a balance between ensuring safety compliance and respecting individual privacy rights is a critical aspect that requires careful consideration and adherence to regulations.

8. Limited Research on Behavior Analysis:

Existing literature suggests a gap in research on combining safety gear surveillance with behavior analysis. Monitoring not only the presence of safety gear but also the adherence to safe practices and behaviors is an area that could contribute to a more holistic safety surveillance approach.

9. Model Generalization Across Industries:

Literature indicates challenges in generalizing YOLO-based safety gear detection models across different industries. Customization may be required to account for industry-specific safety gear requirements, leading to a need for adaptable and industry-agnostic models.

10. Cost and Resource Implications:

Some studies discuss the potential cost and resource implications associated with deploying and maintaining YOLO-based safety gear surveillance systems. Considerations such as hardware requirements, ongoing maintenance, and staff training are important factors that need to be addressed for successful system implementation.

Addressing these existing problems outlined in the literature is crucial for the development of an effective and reliable YOLO-based safety gear surveillance system under the ConstructGuard project. The integration of solutions to these challenges will contribute to the overall success and impact of the safety surveillance system in real-world applications.

Top of Form

* 1. References

1.

S. A. Velastin, B. A. Boghossian, and M. A. Vicencio-Silva, “A motion-based image processing system for detecting potentially dangerous situations in underground railway stations,” Transportation Research Part C: Emerging Technologies, vol. 14, no. 2, pp. 96–113, 2006.

View at: [Publisher Site](https://doi.org/10.1016/j.trc.2006.05.006) | [Google Scholar](https://scholar.google.com/scholar_lookup?title=A%20motion-based%20image%20processing%20system%20for%20detecting%20potentially%20dangerous%20situations%20in%20underground%20railway%20stations&author=S.%20A.%20Velastin&author=B.%20A.%20Boghossian&author=M.%20A.%20Vicencio-Silva&publication_year=2006)

2.

United Nations, Office on Drugs and Crime, Report on “Global Study of Homicide”, <https://www.unodc.org/documents/data-and-analysis/gsh/Booklet1.pdf>.

3.

P. M. Kumar, U. Gandhi, R. Varatharajan, G. Manogaran, R. Jidhesh, and T. Vadivel, “Intelligent face recognition and navigation system using neural learning for smart security in internet of things,” Cluster Computing, vol. 22, no. S4, pp. 7733–7744, 2019.

View at: [Publisher Site](https://doi.org/10.1007/s10586-017-1323-4) | [Google Scholar](https://scholar.google.com/scholar_lookup?title=Intelligent%20face%20recognition%20and%20navigation%20system%20using%20neural%20learning%20for%20smart%20security%20in%20internet%20of%20things&author=P.%20M.%20Kumar&author=U.%20Gandhi&author=R.%20Varatharajan&author=G.%20Manogaran&author=R.%20Jidhesh&author=T.%20Vadivel&publication_year=2019)

4.

V. Babanne, N. S. Mahajan, R. L. Sharma, and P. P. Gargate, “Machine learning based smart surveillance system,” in Proceedings of the 2019 Third International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC), pp. 84–86, IEEE, Palladam, India, December 2019.

View at: [Google Scholar](https://scholar.google.com/scholar_lookup?title=Machine%20learning%20based%20smart%20surveillance%20system&author=V.%20Babanne&author=N.%20S.%20Mahajan&author=R.%20L.%20Sharma&author=P.%20P.%20Gargate)

5.

A. Joshi, N. Jagdale, R. Gandhi, and S. Chaudhari, “Smart surveillance system for detection of suspicious behaviour using machine learning,” in Intelligent Computing, Information and Control Systems. ICICCS 2019. Advances in Intelligent Systems and Computing, A. Pandian, K. Ntalianis, and

R. Palanisamy, Eds., vol. 1039, Springer, Cham, Berlin, Germany, 2020.

View at: [Publisher Site](https://doi.org/10.1007/978-3-030-30465-2_2) | [Google Scholar](https://scholar.google.com/scholar_lookup?title=Smart%20surveillance%20system%20for%20detection%20of%20suspicious%20behaviour%20using%20machine%20learning&author=A.%20Joshi&author=N.%20Jagdale&author=R.%20Gandhi&author=S.%20Chaudhari&publication_year=2020)

* 1. Problem Statement Definition

1.

Write automated test scripts, which ensures that each and every section or a unit meets its design and behaves as expected. Create comprehensive unit tests and mention the categorization into method/class, function and module. Document the description of tests and possible test outcomes. Write sufficient test coverage for all the scenarios, the code can be eliminated. The application is developed using Python Flask RESTPLus. Check the resources URLs to find more about the technology.

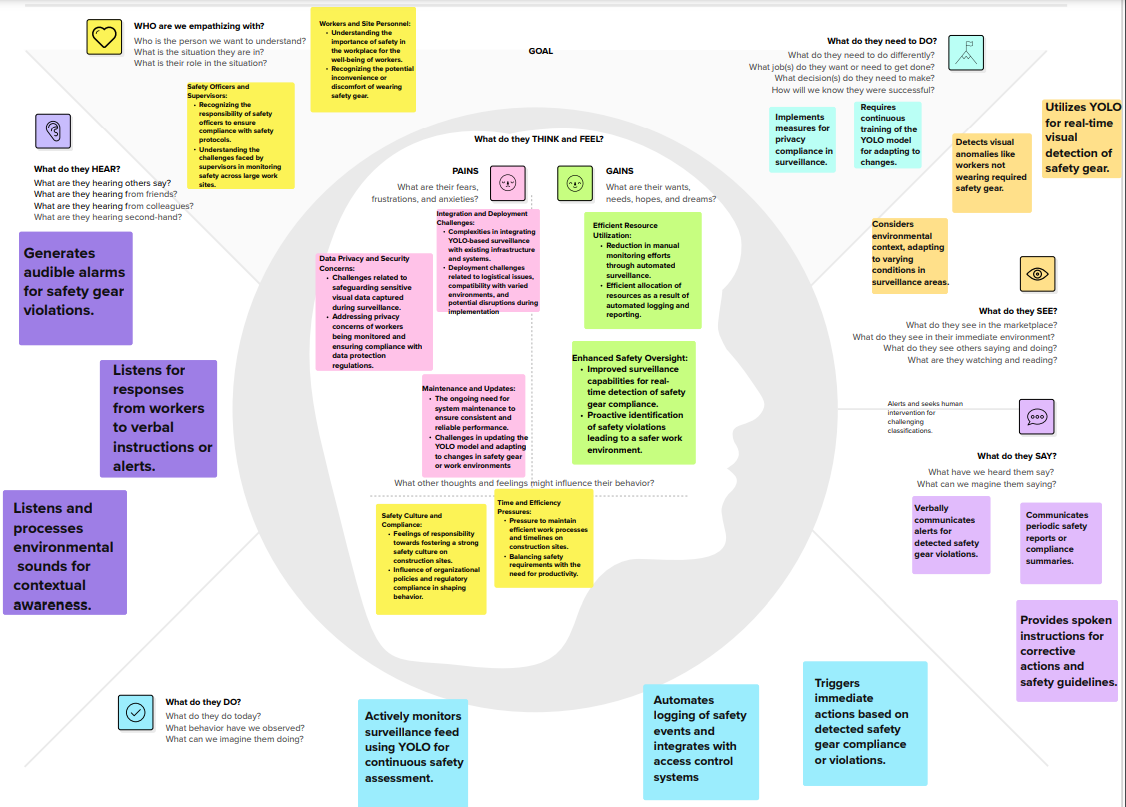
2.

NIC is responsible for creating highly scalable application stacks. Some of the application stacks are widely used throughout the organization in multiple projects. The objective of this task is to create generic application catalogues for scalable application stacks. Teams can use the orchestration languages as per their choice for Containers or VM deployments. The preferable languages are OpenStack HEAT, AWS Cloud Formation or HELM for Kubernetes. Some of the example stacks can be popular applications like: Drupal + Redis + MariaDB + S3fs, WordPress + Redis + MariaDB + S3fs, Elastic + Logstash + Kibana. This process would also help migrate legacy applications to new stacks. Proposed Approach for Implementation of Elastic Stack “Elastic + Logstash + Kibana” & Suggestions for Including SIEM (Security Information & Event Management), APM (Application Performance Monitoring), ML (Machine Learning) for your Infrastructure Anomaly Detection with the Elastic Stack.

# IDEATION & PROPOSED SOLUTION

* 1. Empathy Map Canvas

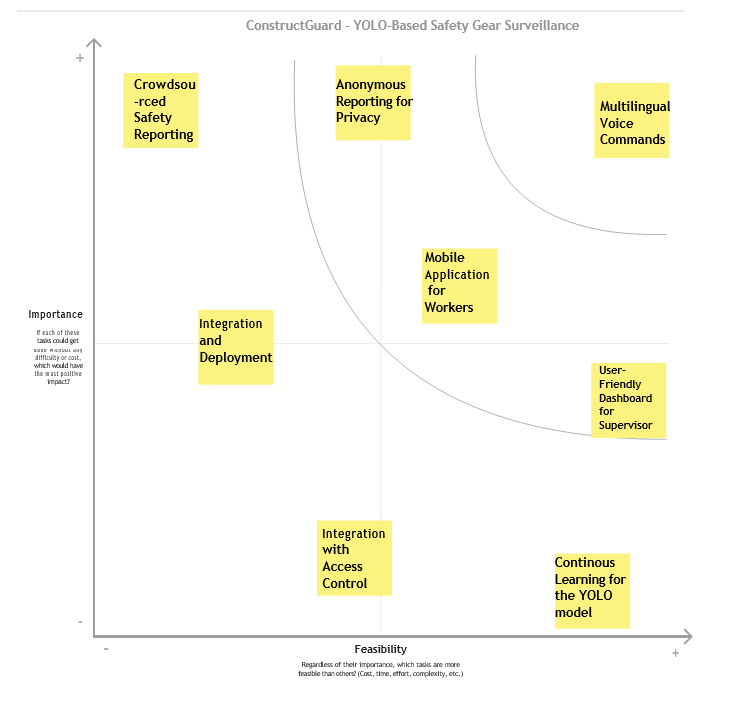
An empathy map is a tool used to understand the thoughts, feelings, actions, and needs of users or stakeholders. In the context of a YOLO-Based Safety Gear Surveillance System, here's an empathy map canvas

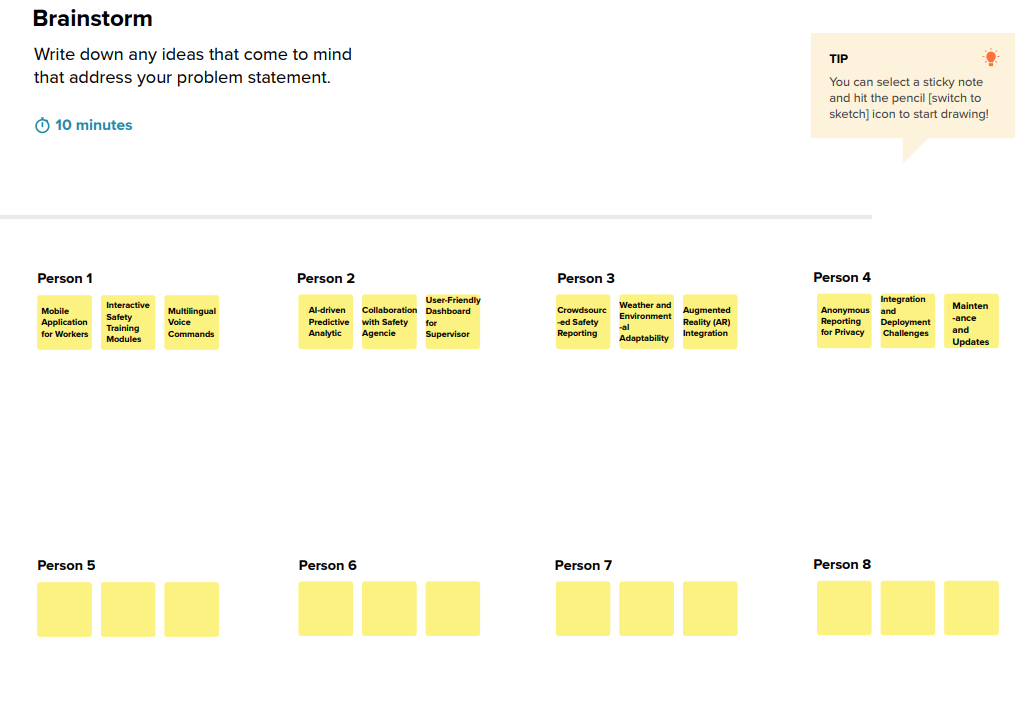


* 1. Ideation & Brainstorming

Step 2

Ideation and brainstorming are creative processes to generate a wide range of ideas. Here's a brainstorming session for the YOLO-Based Safety Gear Surveillance System:



Step 1 

# REQUIREMENT ANALYSIS

* 1. Functional requirement

These functional requirements focus on the core capabilities necessary for an effective and user-friendly YOLO-Based Safety Gear Surveillance System. They encompass real-time detection, alerting, privacy considerations, integration, user interface design, and continuous learning for improved accuracy.

1.Real-Time Object Detection: The system must employ YOLO-based object detection to identify and classify safety gear items in real-time.

2. Alert System : Provide a robust alert system to notify relevant personnel or safety officers in case of safety gear non-compliance.

3.Privacy-Preserving Features: Ensure the system respects privacy by implementing features that protect sensitive data.

4 Integration with Access Control: Optionally integrate the system with access control mechanisms to restrict entry for non-compliant individuals

5 User-Friendly Interface: Develop an intuitive and user-friendly interface for system monitoring and management.

* 1. :Non-Functional requirements

Performance:

Description: The system must demonstrate high-performance capabilities to handle real-time processing and detection.

Scalability:

Description: The system should be scalable to accommodate varying numbers of surveillance cameras and adapt to the size of the monitored area.

Reliability:

Description: Ensure a high level of reliability to minimize system downtime and false alarms.

Security:

Description: Implement robust security measures to protect the integrity of the system and the privacy of captured data

Usability:

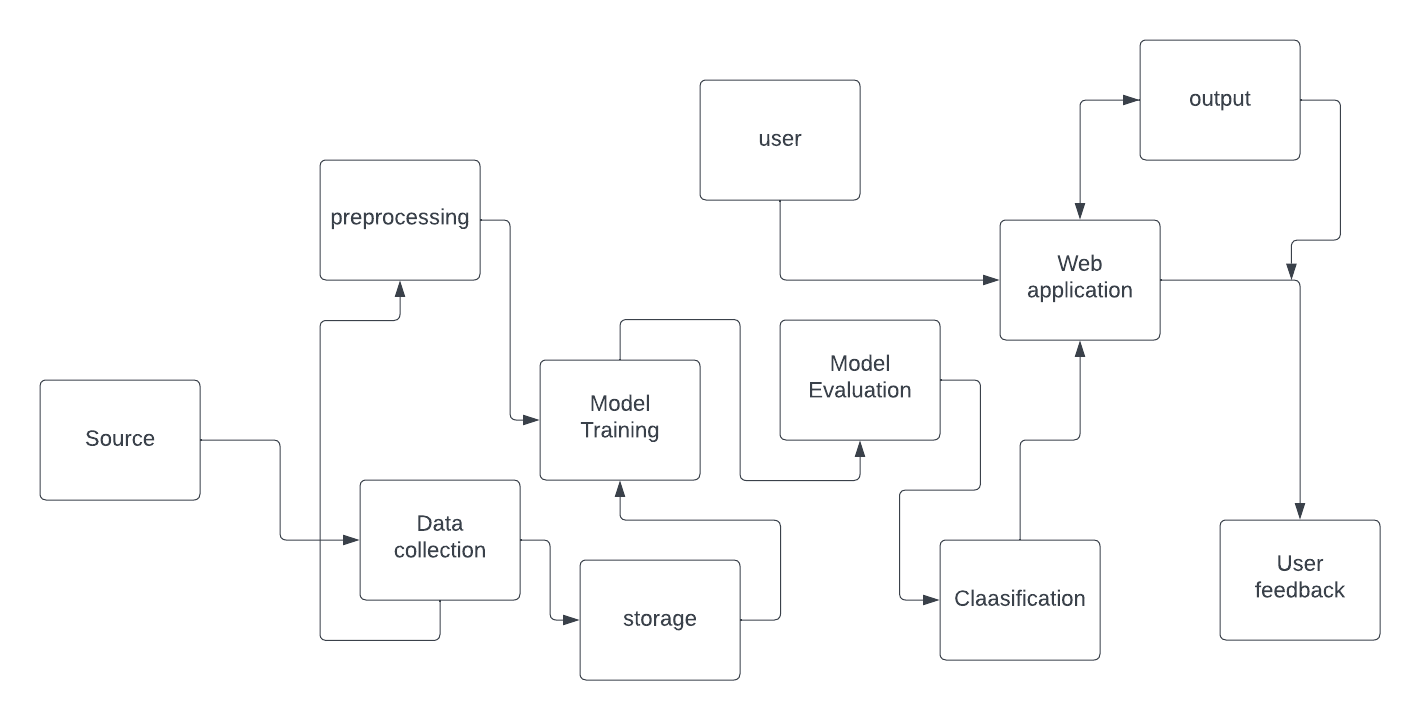
Description: Design the system with a focus on usability to ensure that users can easily navigate and interact with the interface.

# PROJECT DESIGN

* 1. Data Flow Diagrams & User Stories

Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



User Stories

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| User Type | Functional Requirement (Epic) | User Story Number | User Story / Task | Acceptance criteria | Priority | Release |
| Construction Manager | Monitor construction site for safety violations | US001 | As a construction manager, I want to get alerted to unsafe conditions on site so I can quickly address them | System detects common safety violations like lack of hard hats, harnesses, improper equipment use etc.  Alerts are sent to construction manager in real-time via email/text with screenshot and details. | High | v1 |
| Construction Worker | Request help in case of emergency | US002 | As a construction worker, I want to be able to quickly request help in case of an emergency so I can get assistance faster. | Workers are provided a safety button that can be pressed to trigger emergency alert.  Alert sent to safety manager with worker's location.  Emergency response process is initiated immediately. | High | V1 |
| Safety Manager | Generate safety compliance reports | US003 | As a safety manager, I want to be able to pull up-to-date compliance reports so I can identify risks and demonstrate adherence. | Reports on safety violations, trends, and metrics available on demand.  Reports meet regulatory compliance requirements.  Data exportable to share with stakeholders. | Medium | V2 |

* 1. Solution Architecture

The key components of the architecture include data collection consist of reviewing the dataset and preprocessing, model selection leveraging convolutional neural networks (CNNs) and transfer learning from pre-trained models, custom model architecture design, training with appropriate loss functions and metrics, hyperparameter tuning, deployment to a production environment, integration with a user interface, and ongoing monitoring and maintenance. The goal is to leverage deep learning and computer vision techniques to accurately classify images into different disease categories. A robust, well-designed architecture allows for adaptability over time as new data becomes available.

Our solution leverages Convolutional Neural Networks (CNNs) to address the garbage classification

problem effectively.

● Data Gathering

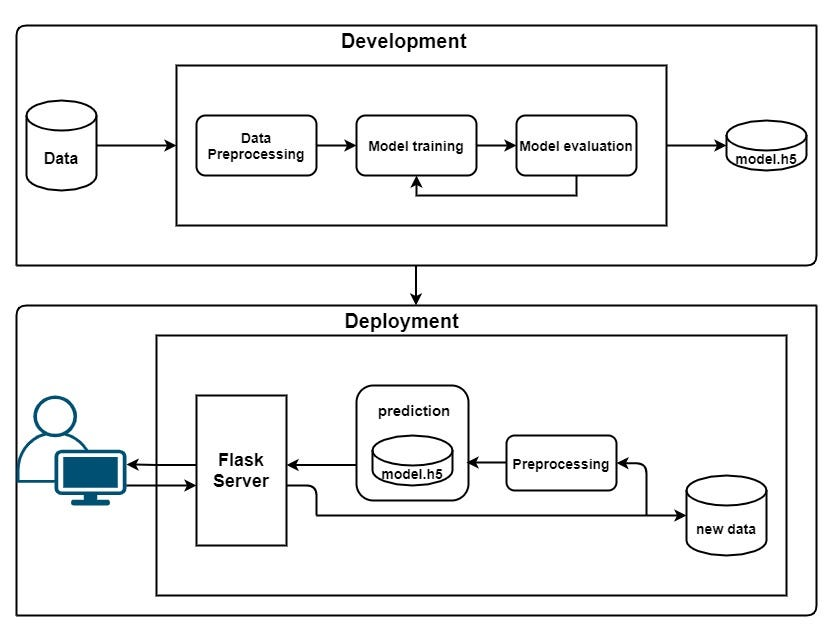
● Image Preprocessing

● Model Building

● classification/Prediction

● Real Time Analysis

Example - Solution Architecture Diagram:

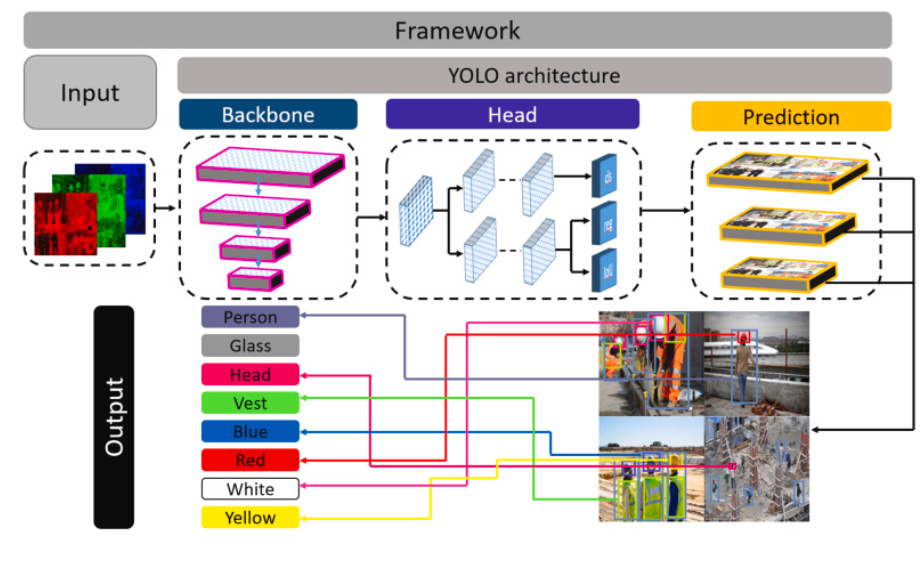


# PROJECT PLANNING & SCHEDULING

6.1Technical Architecture

Construct Guard:

YOLO-Based Safety Gear Surveillance" stands as an innovative application, seamlessly merging computer vision and artificial intelligence to elevate safety and security within construction sites. The backbone of this cutting-edge system lies in the implementation of YOLO (You Only Look Once), a state-of-the-art object detection algorithm. This robust technology excels in precisely identifying and verifying the presence of crucial safety gear donned by construction workers.



* 1. Sprint Planning & Estimation

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1. **Sprint** | **Functional Requirement**  **(Epic)** | **User Story Number** | **User Story / Task** | **Story Points** | **Priority** | **Team Members** |
| Sprint-1 | Project setup& Infrastructure | USN-1 | Set up the development environment with the required tools and frameworks to predict the eye disease. | 1 | High | SAI KAMAL |
| Sprint-1 | development environment | USN-2 | Gather a diverse dataset of images containing different types of eye diseases for training the deep learning model. | 2 | High | KAMAL |
| Sprint-2 | Data collection | USN-3 | Preprocess the collected datasetby resizing images,normalizing pixel values, and splitting it into training and validation sets. | 2 | High | RAKESH |
| Sprint-2 | data preprocessing | USN-4 | Explore and evaluate different deep learning architectures (e.g., CNNs, Transfer learning) to select the most suitable model for garbage classification. | 3 | High | RAKESH |
| Sprint-3 | model development | USN-5 | train the selected deep learning modelusing the preprocessed dataset and monitorits performance on the validation set. | 4 | High | NAZZIIA |
| Sprint-3 | Training | USN-6 | implement data augmentation techniques (e.g., rotation, flipping)to improve the model's robustness and accuracy. | 6 | medium | NAZZIIA |
| Sprint-4 | model deployment & Integration | USN-7 | deploy the trained deep learning model as an API or web service to make it accessible for common people. integrate the model's API into a user-friendly web interface for users to upload images and get the accurate results. | 1 | medium | SHISRSHA |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Sprint-5 | Testing & quality  assurance | USN-8 | conduct thorough testing of the  model and web interface to identify and report any issues or bugs. fine-tune the model hyperparameters and optimize its performance basedon user feedback and testing results. | 1 | medium | SHIRISHA |

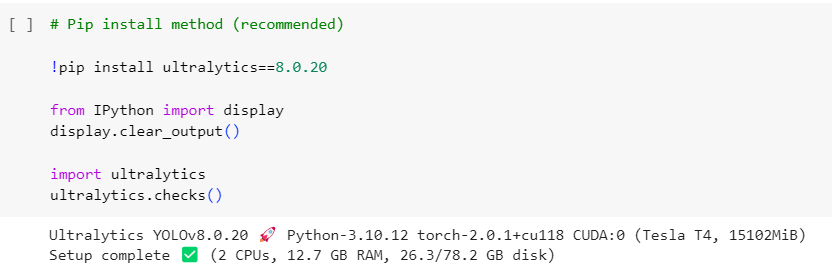
* 1. Sprint Delivery Schedule

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1. **Sprint** | **Total Story Points** | **Duration** | **Sprint Start Date** | **Sprint End Date (Planned)** | **Story Points Completed (as on Planned End Date)** | **Sprint Release Date (Actual)** |
| Sprint-1 | 3 | 3 Days | 25 oct 2023 | 27 oct 2023 | 3 | 27 oct 2023 |
| Sprint-2 | 5 | 3 Days | 27 oct 2023 | 01 Nov 2023 | 5 |  |
| Sprint-3 | 10 | 5 Days | 01 Nov 2023 | 05 Nov 2023 | 10 |  |
| Sprint-4 | 1 | 7 Days | 05 Nov 2023 | 12 Nov 2023 | 1 |  |
| Sprint-5 | 1 | 7 Days | 13 Nov 2023 | 18 Nov 2023 | 1 |  |

# CODING & SOLUTIONING (Explain the features added in the project along with code)

* 1. Feature 1

Data collection

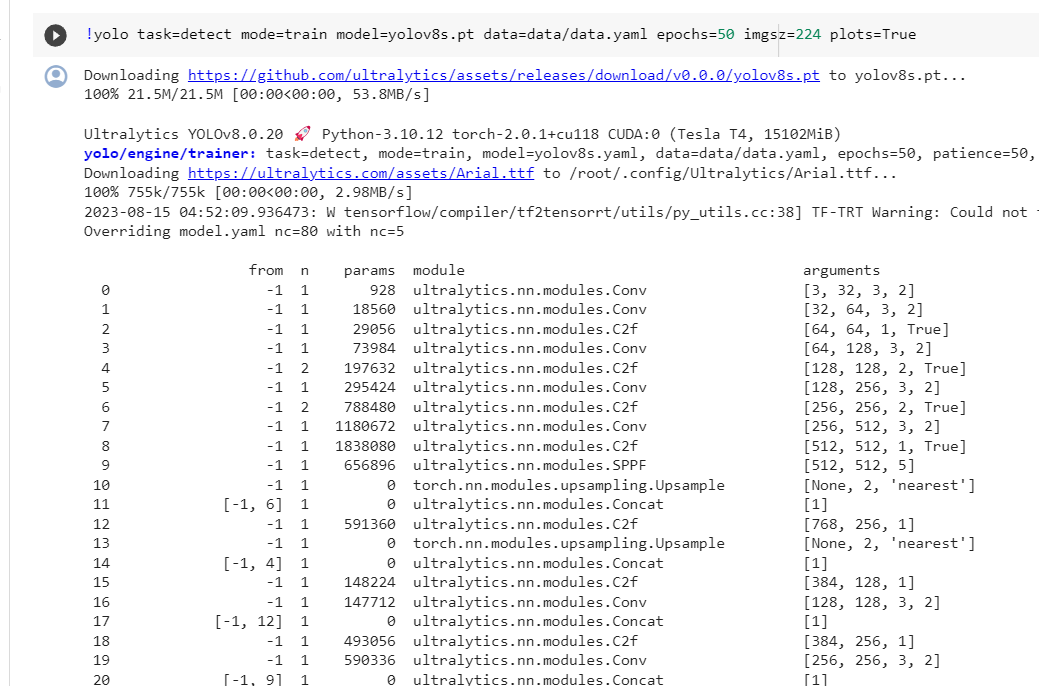


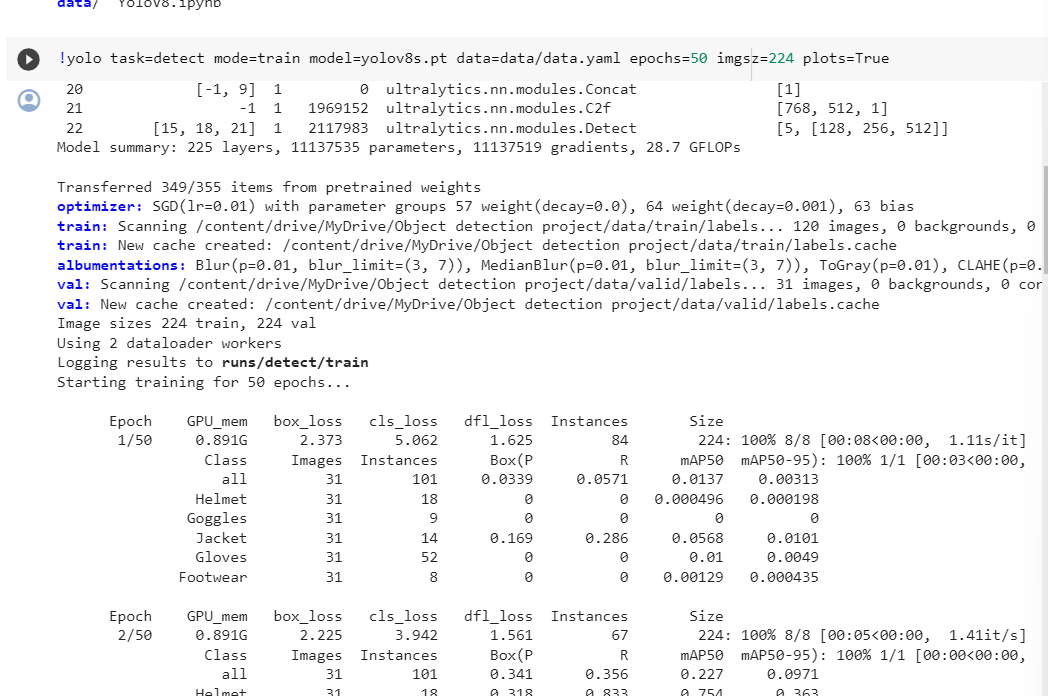


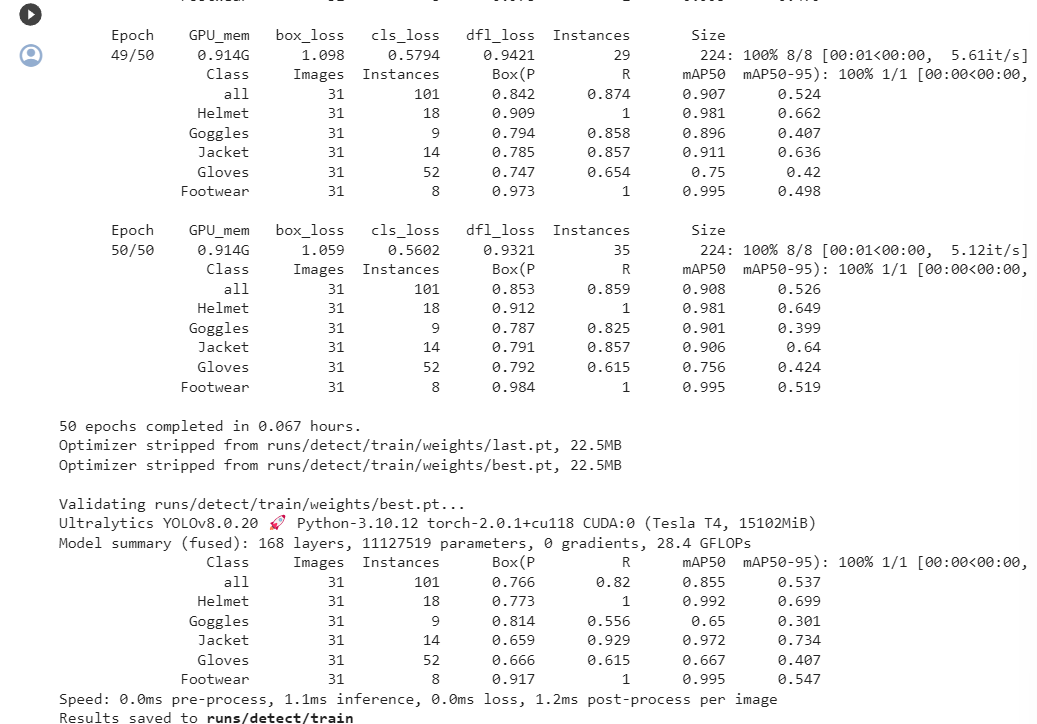
Assigning the input image size



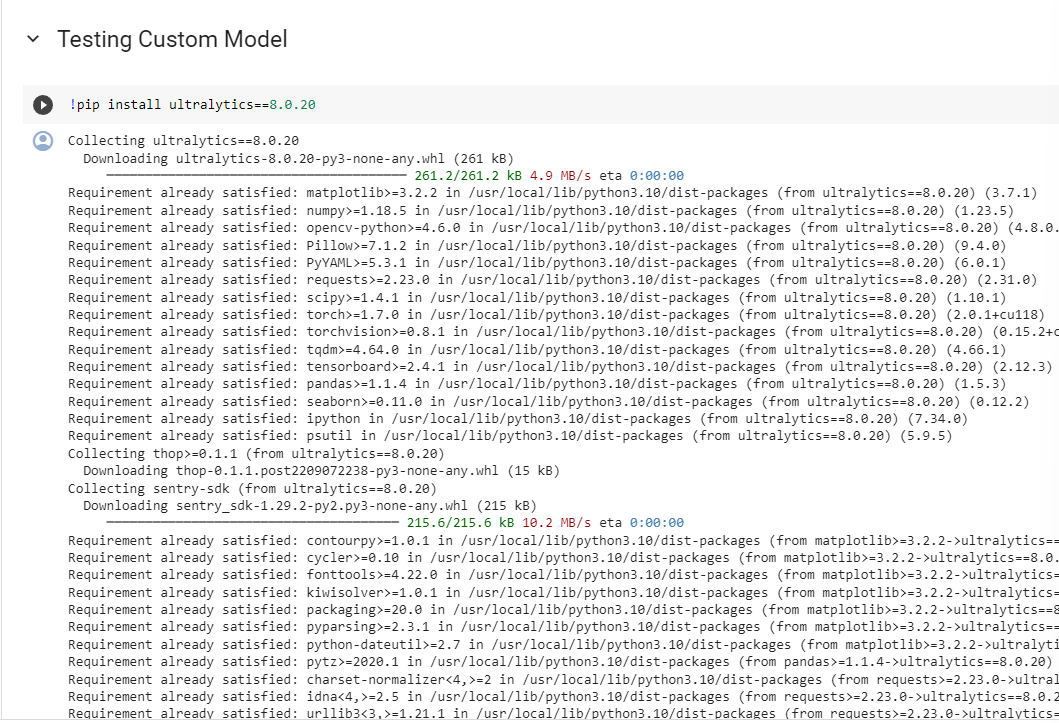


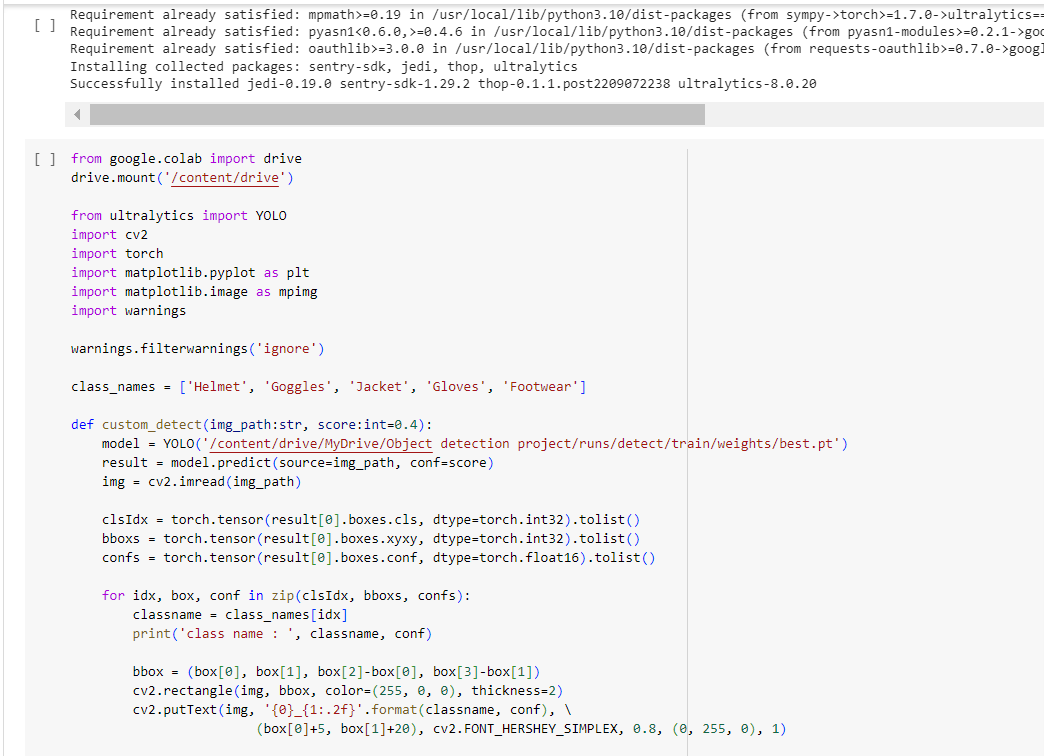


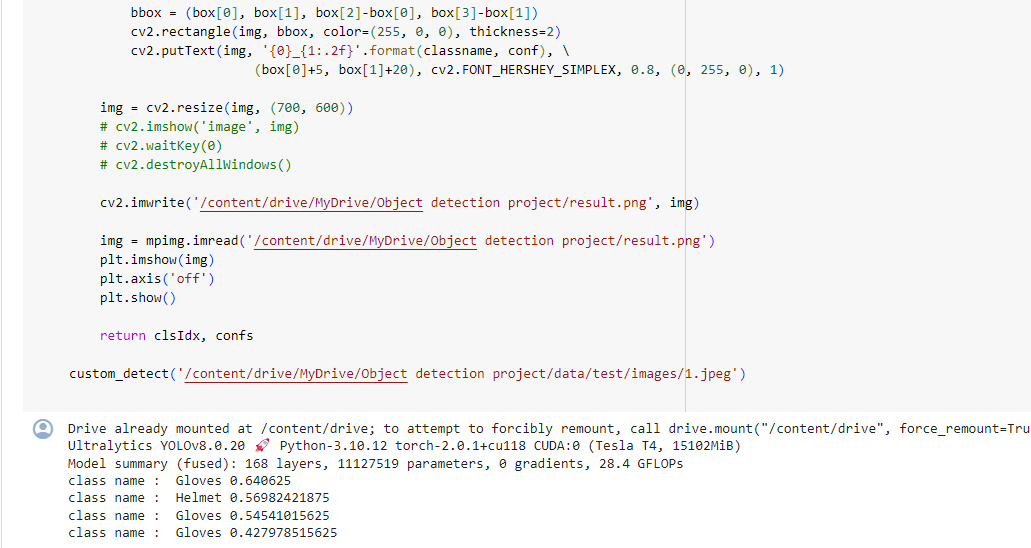


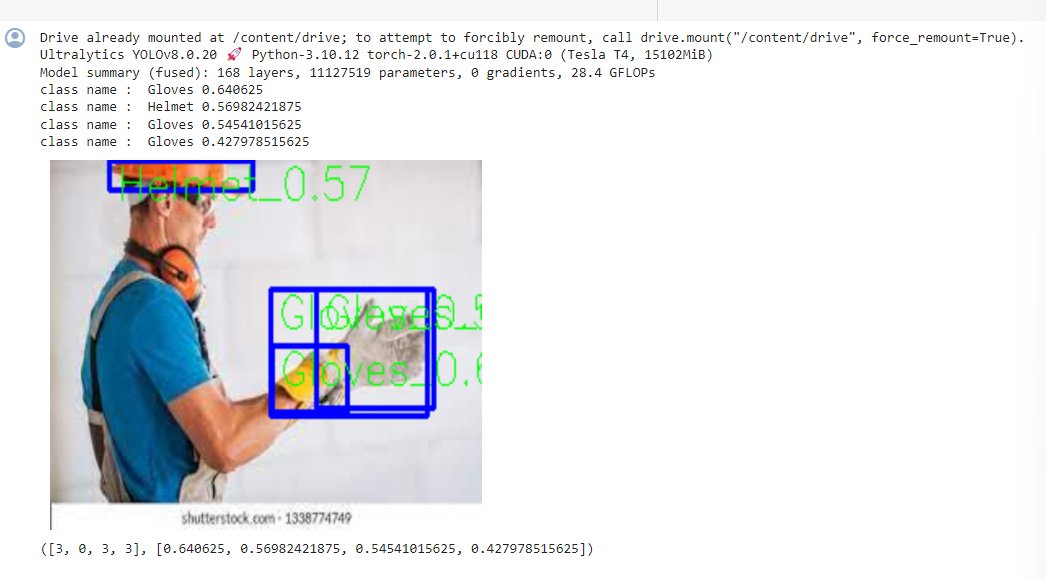








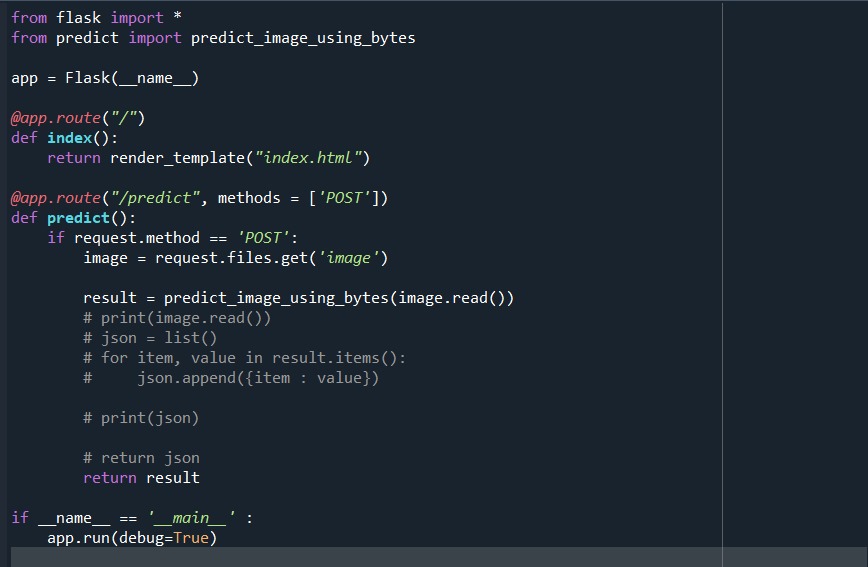




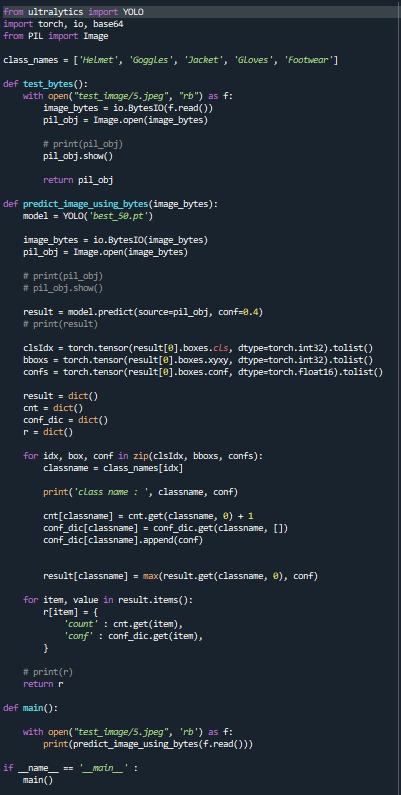
­

* 1. Feature2

App.py file

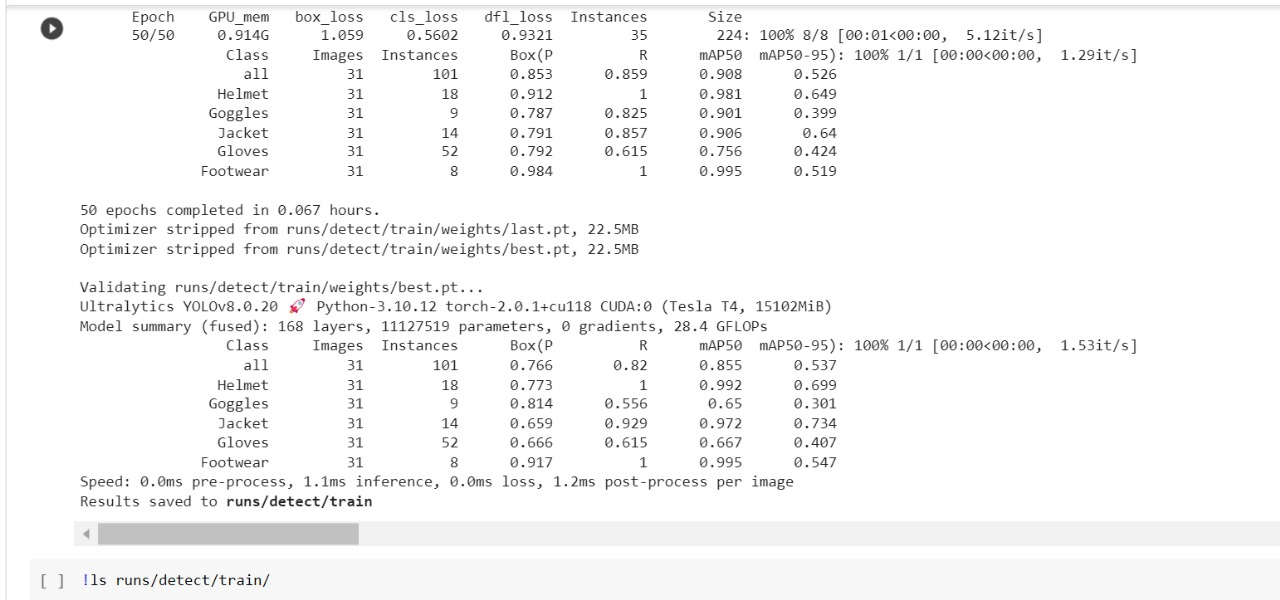


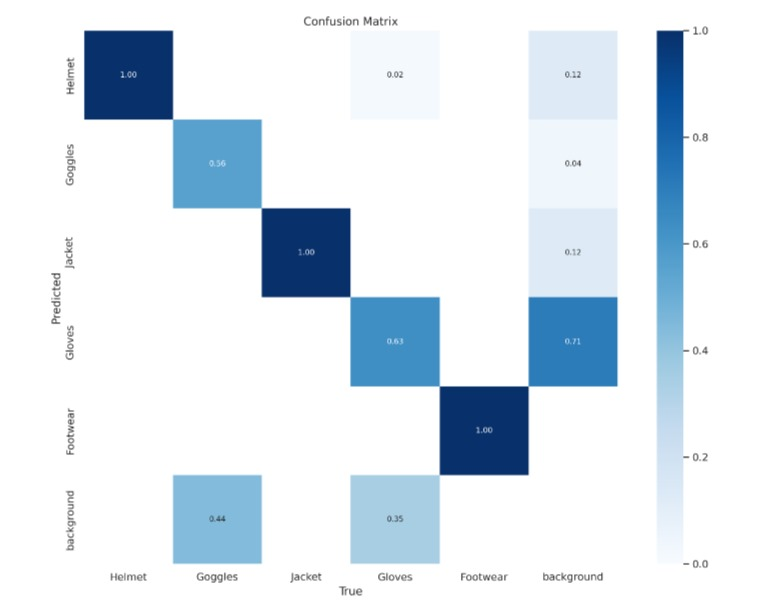
Predict.py file



# PERFORMANCE TESTING

* 1. Performace Metrics







# RESULTS

* 1. Output Screenshots

# 

# 

# 

# 

# 

# ADVANTAGES :

1. Real-Time Detection: YOLO enables real-time object detection, allowing immediate identification of safety gear compliance or violations.

2. Efficiency in Processing: YOLO's single-pass approach increases processing speed, making it efficient for monitoring multiple video feeds simultaneously.

3. Accuracy and Precision: YOLO provides accurate and precise detection of safety gear items, reducing false positives and negatives.

4. Adaptability to Various Environments: YOLO-based systems can adapt to different lighting conditions and environmental variations, enhancing versatility.

5. User-Friendly Interface: YOLO-Based Safety Gear Surveillance Systems can be integrated with user-friendly interfaces, making it accessible for both administrators and end-users.

6. Integration with Access Control: YOLO systems can seamlessly integrate with access control mechanisms, enhancing overall security protocols.

7. Continuous Learning and Improvement: YOLO systems support continuous learning and model improvement, adapting to evolving safety standards.

8. Scalability: YOLO-based systems can scale horizontally to accommodate additional cameras and expanded monitoring areas.

9. Privacy-Preserving Features: YOLO systems can incorporate privacy-preserving features, addressing concerns related to constant surveillance.

10. Cost-Effective Deployment: YOLO systems can be cost-effective to deploy, especially when compared to traditional surveillance systems.

# DISADVANTAGES:

1. Limited Dataset Variability: YOLO systems may face challenges due to a limited dataset for training, impacting adaptability to diverse scenarios.

2. Occlusion Handling: YOLO systems may struggle to handle occlusions, affecting the accuracy of safety gear detection in crowded or obstructed environments.

3. Resource Intensive: YOLO systems can be resource-intensive, requiring powerful hardware for optimal performance.

4. Ethical and Privacy Concerns: Constant surveillance raises ethical and privacy concerns, requiring careful consideration and communication.

5. Integration Challenges: Integrating YOLO systems with existing infrastructure, especially in legacy systems, may present challenges.

6. Dependency on Training Data: The performance of YOLO models heavily depends on the quality and representativeness of the training dataset.

7. False Positives and Negatives: YOLO systems may still encounter challenges in minimizing false positives and negatives, impacting the reliability of safety alerts.

8. Potential for System Bias: If not properly curated, training data may introduce biases, leading to discriminatory outcomes in safety gear detection.

9. Complex Implementation for Behavior: Analysis Integrating behavior analysis features may add complexity to the system, requiring careful design and testing.

10. Initial Learning Curve: Implementing a YOLO-based system may involve a learning curve for administrators and users unfamiliar with the technology.

1. **CONCLUSION**

The YOLO-Based Safety Gear Surveillance System presents a promising solution for enhancing workplace safety in high-risk environments. Through its real-time object detection capabilities and efficiency in processing multiple video feeds, YOLO brings several advantages to safety monitoring systems. The system's accuracy and precision in identifying safety gear items contribute to minimizing false positives and negatives, providing a reliable means of ensuring compliance with safety protocols.

The adaptability of YOLO to various environments and its continuous learning features make it well-suited for dynamic and evolving workplace conditions. The user-friendly interfaces and seamless integration with access control mechanisms further enhance its usability and contribute to a comprehensive approach to workplace security.

However, it's essential to acknowledge the challenges associated with YOLO-based systems. These include potential limitations in dataset variability, occlusion handling, and the need for powerful hardware resources. Additionally, ethical considerations regarding constant surveillance and privacy concerns must be addressed to ensure user acceptance and compliance.

In conclusion, the YOLO-Based Safety Gear Surveillance System represents a powerful tool for promoting workplace safety, with its strengths lying in real-time detection, adaptability, and user-friendly features. As technology advances and these challenges are mitigated, YOLO systems are poised to play a crucial role in creating safer working environments, reducing accidents, and fostering a culture of adherence to safety protocols. The ongoing refinement and adaptation of YOLO models will further contribute to the effectiveness and acceptance of safety surveillance systems in various industries.

Top of Form

# FUTURE SCOPE

The future scope of the YOLO-Based Safety Gear Surveillance System holds significant potential for further advancements in workplace safety, technology integration, and user experience. Here are several avenues for future development:

1. Enhanced Object Recognition:

Future Scope:

Continue research and development to improve the accuracy and efficiency of safety gear detection. Explore advanced neural network architectures to achieve even higher precision, especially in challenging scenarios such as crowded environments.

2. Multi-Sensor Integration:

Future Scope:

Integrate YOLO-based systems with a variety of sensors beyond traditional surveillance cameras. This could include thermal imaging, LiDAR, or wearable sensors, providing a more comprehensive safety monitoring solution.

3. Edge Computing Implementation:

Future Scope:

Investigate the implementation of edge computing to perform real-time processing directly on the surveillance cameras or edge devices. This can reduce the need for centralized processing and enhance system responsiveness.

4. Behavior Analytics Refinement:

Future Scope:

Evolve the system's behavior analytics capabilities to not only detect safety gear but also analyze and improve overall workplace safety practices. This could involve identifying risky behaviors and providing targeted interventions for better safety outcomes.

5. Explainable AI (XAI):

Future Scope:

Integrate explainable AI features into the system to provide clear insights into how safety gear compliance decisions are made. This can enhance transparency, user trust, and facilitate easier auditing of the system's decisions.

6. Human-Centric Design:

Future Scope:

Focus on human-centric design principles to make the system more intuitive and user-friendly. Conduct user experience studies and gather feedback to refine interfaces and enhance the overall usability of the safety surveillance system.

7. Automated Maintenance and Health Monitoring:

Future Scope:

Implement automated maintenance and health monitoring features for the YOLO-based system. This could involve self-diagnostic capabilities, predictive maintenance, and automated updates to ensure optimal system performance.

8. Customization for Industry-Specific Needs:

Future Scope:

Develop customization options to tailor the YOLO-Based Safety Gear Surveillance System to specific industry requirements. This could involve adapting the system to recognize industry-specific safety gear and compliance standards.

9. Integration with AI-Powered Safety Training:

Future Scope:

Explore integration with AI-powered safety training platforms. Use the surveillance system's data to identify common safety lapses and incorporate these insights into training modules to enhance overall safety awareness.

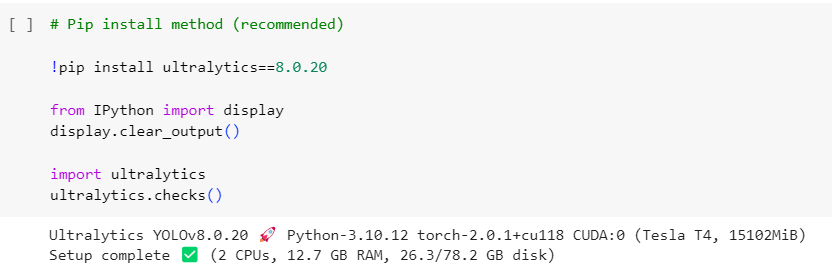
10. Blockchain for Data Integrity:

Future Scope:

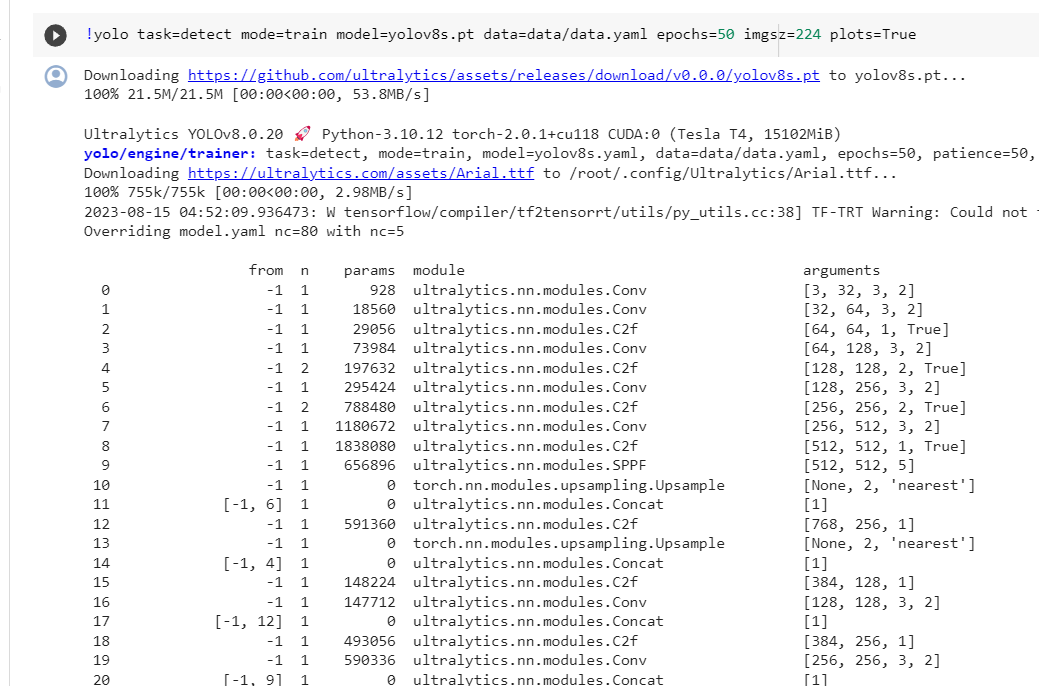
Investigate the use of blockchain technology to ensure the integrity and security of data collected by the YOLO-based system. This can enhance data trustworthiness and provide a tamper-resistant record of safety compliance

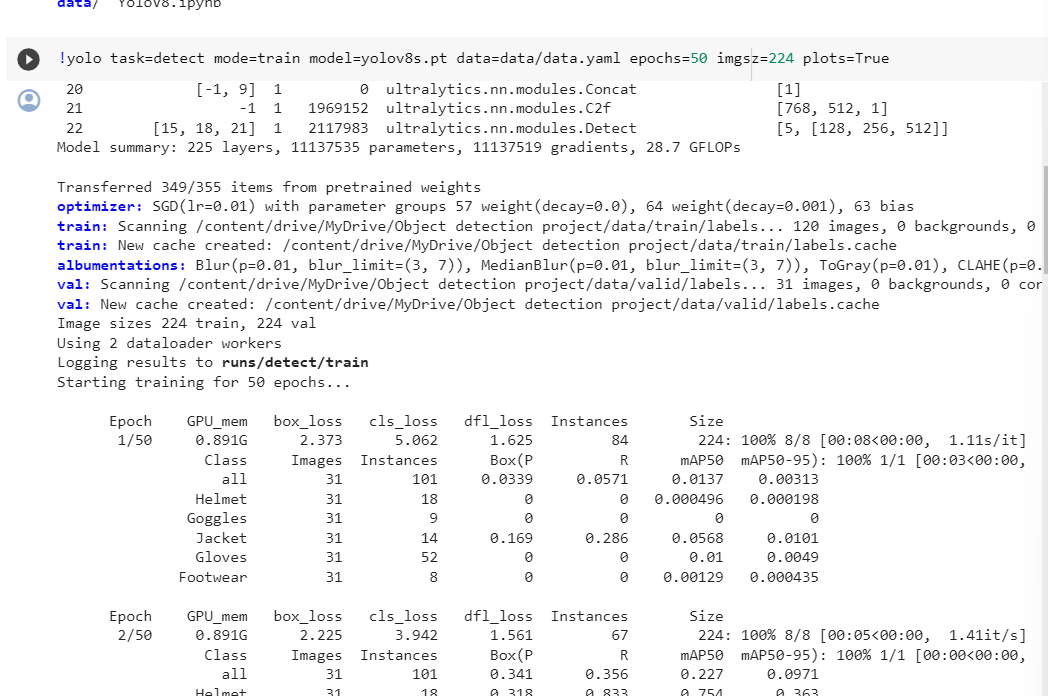
1. **APPENDIX**

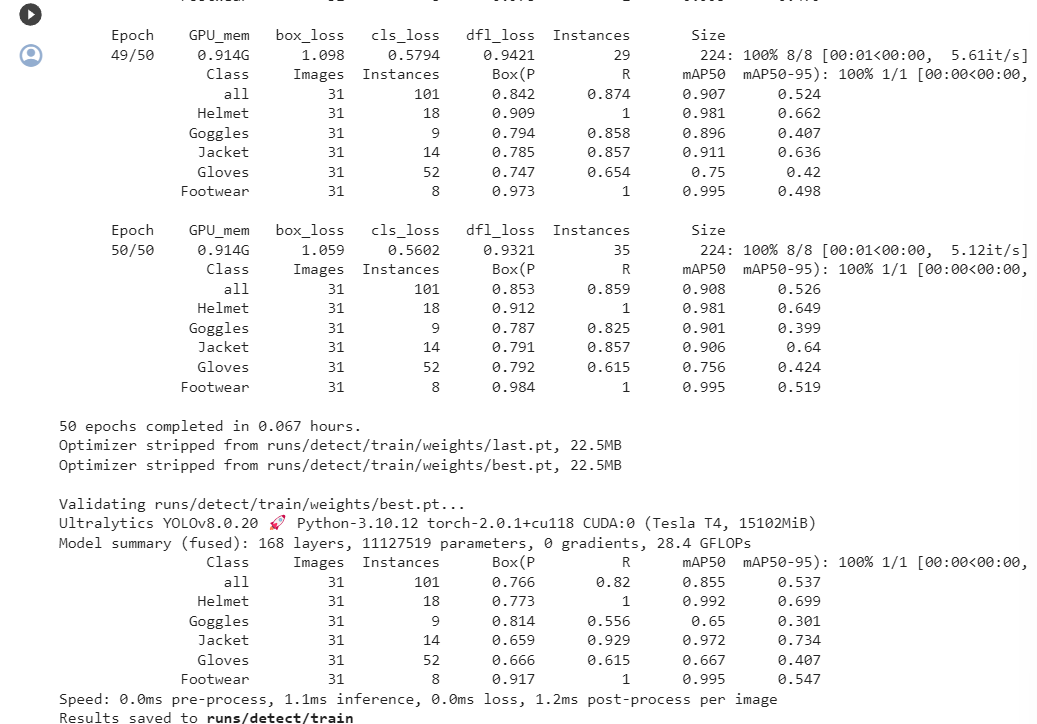
Source Code



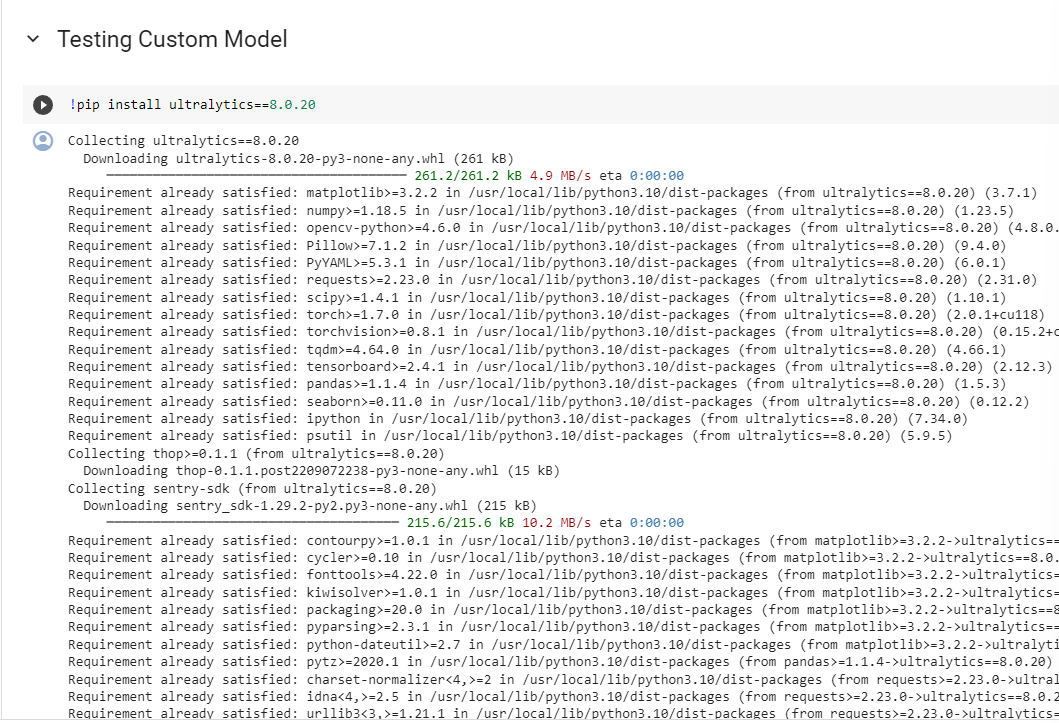


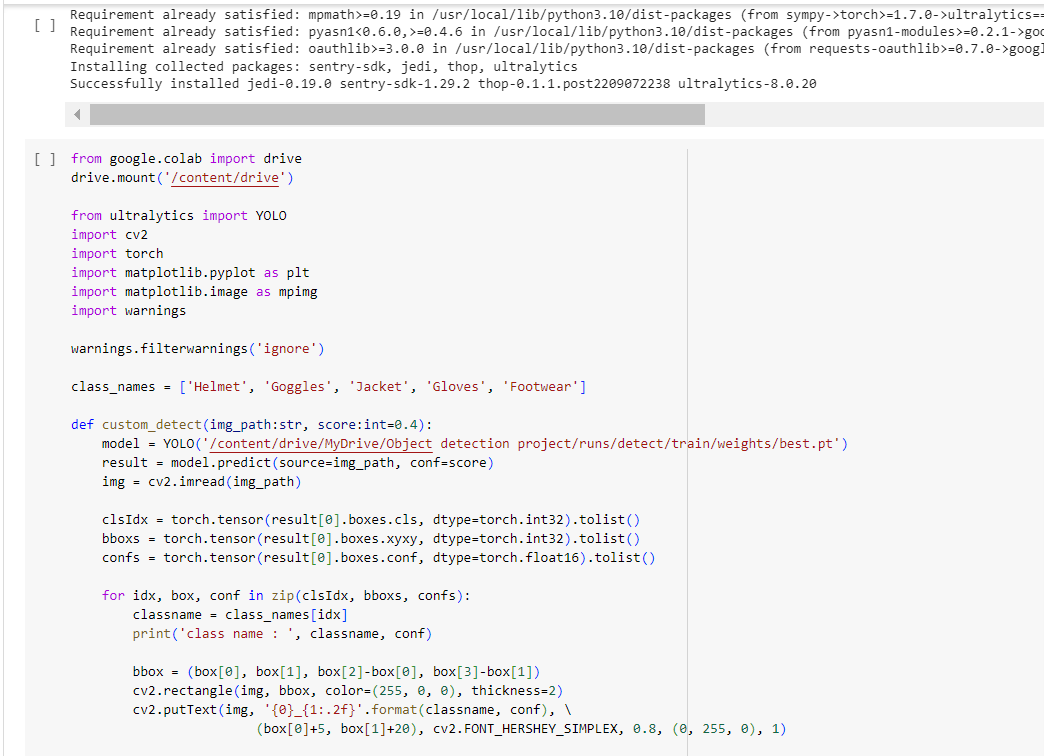


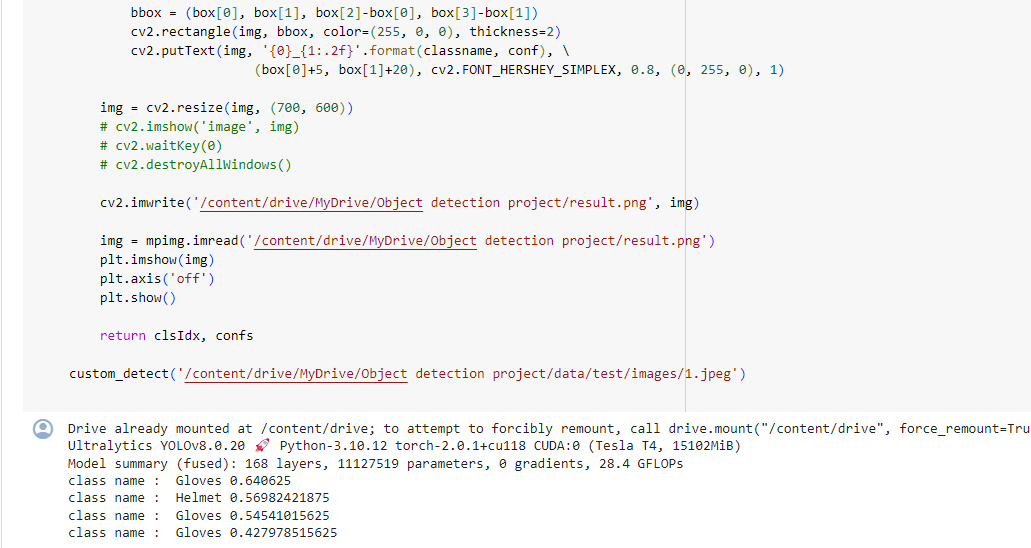


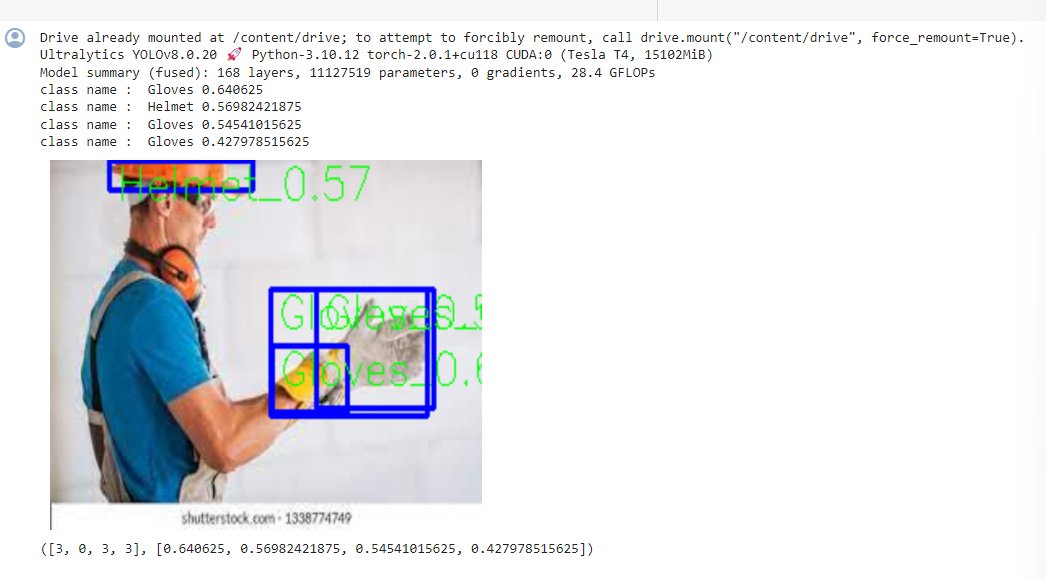












Source Code

!nvidia-smi

from google.colab import drive

drive.mount('/content/drive')

import os

HOME = os.getcwd()

print(HOME)

# Pip install method (recommended)

!pip install ultralytics==8.0.20

from IPython import display

display.clear\_output()

import ultralytics

ultralytics.checks()

from ultralytics import YOLO

from IPython.display import display, Image

%cd {HOME}

!yolo task=detect mode=predict model=yolov8n.pt conf=0.25 source='https://media.roboflow.com/notebooks/examples/dog.jpeg'

model = YOLO(f'{HOME}/yolov8n.pt')

results = model.predict(source='https://media.roboflow.com/notebooks/examples/dog.jpeg', conf=0.25)

results[0].boxes.xyxy

results[0].boxes.conf

results[0].boxes.cls

%cd /content/drive/MyDrive/Object detection project

%cd /content/drive/MyDrive/Object detection project

%ls

!yolo task=detect mode=train model=yolov8s.pt data=data/data.yaml epochs=50 imgsz=224 plots=True

!ls runs/detect/train/

Image(filename='runs/detect/train/confusion\_matrix.png', width=600)

Image(filename='runs/detect/train/results.png', width=600)

Image(filename='runs/detect/train/val\_batch0\_pred.jpg', width=600)

!pip install ultralytics==8.0.20

from google.colab import drive

drive.mount('/content/drive')

from ultralytics import YOLO

import cv2

import torch

import matplotlib.pyplot as plt

import matplotlib.image as mpimg

import warnings

warnings.filterwarnings('ignore')

class\_names = ['Helmet', 'Goggles', 'Jacket', 'Gloves', 'Footwear']

def custom\_detect(img\_path:str, score:int=0.4):

model = YOLO('/content/drive/MyDrive/Object detection project/runs/detect/train/weights/best.pt')

result = model.predict(source=img\_path, conf=score)

img = cv2.imread(img\_path)

clsIdx = torch.tensor(result[0].boxes.cls, dtype=torch.int32).tolist()

bboxs = torch.tensor(result[0].boxes.xyxy, dtype=torch.int32).tolist()

confs = torch.tensor(result[0].boxes.conf, dtype=torch.float16).tolist()

for idx, box, conf in zip(clsIdx, bboxs, confs):

classname = class\_names[idx]

print('class name : ', classname, conf)

bbox = (box[0], box[1], box[2]-box[0], box[3]-box[1])

cv2.rectangle(img, bbox, color=(255, 0, 0), thickness=2)

cv2.putText(img, '{0}\_{1:.2f}'.format(classname, conf), \

(box[0]+5, box[1]+20), cv2.FONT\_HERSHEY\_SIMPLEX, 0.8, (0, 255, 0), 1)

img = cv2.resize(img, (700, 600))

# cv2.imshow('image', img)

# cv2.waitKey(0)

# cv2.destroyAllWindows()

cv2.imwrite('/content/drive/MyDrive/Object detection project/result.png', img)

img = mpimg.imread('/content/drive/MyDrive/Object detection project/result.png')

plt.imshow(img)

plt.axis('off')

plt.show()

return clsIdx, confs

custom\_detect('/content/drive/MyDrive/Object detection project/data/test/images/1.jpeg')

GitHub & Project Demo Link

Github link:

https://github.com/smartinternz02/SI-GuidedProject-615770-1700287331

Video link:

https://drive.google.com/file/d/1hbc0RWAxWN5\_z51UVlhp5a9QINy9t3No/view?usp=drivesdk