Detection of Personal Protective Equipment (PPE) Using Deep Learning Algorithm

MIGUEL, M, LACANIENTA*

Mapúa University, mmlacanienta@mymail.mapua.edu.ph

ABSTRACT

This research explores the application of advanced deep learning algorithms, specifically the YOLOv9 architecture, for the automated detection of Personal Protective Equipment (PPE) in hazardous work environments. The study addresses significant gaps in current PPE detection methods, which often face limitations in accuracy and adaptability under variable environmental conditions. Leveraging a custom dataset that includes diverse real-world scenarios, we refine YOLOv9 to improve generalization and computational efficiency in detecting critical safety gear, such as hard hats, across challenging settings. Key contributions include enhancements in model robustness through targeted data augmentation, hyperparameter optimization, and fine-tuning. Experimental results demonstrate a high detection accuracy and rapid processing, showcasing the model's potential for real-time monitoring in industrial applications. The proposed framework holds implications for improving workplace safety compliance and reducing accidents in dynamic environments.

CCS CONCEPTS • Object detection; • Machine learning approaches; • Neural networks; • Computer vision applications; •

Additional Keywords and Phrases: Personal Protective Equipment (PPE), YOLOv9, Deep Learning, Object Detection, Workplace Safety

ACM Reference Format:

Miguel M. Lacanienta. 2024. *Detection of Personal Protective Equipment (PPE) Using Deep Learning Algorithm*: ACM Conference Proceedings Manuscript Submission Template: This is the subtitle of the paper, this document both explains and embodies the submission format for authors using Word. In Woodstock '18: ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY. ACM, New York, NY, USA, 10 pages. NOTE: This block will be automatically generated when manuscripts are processed after acceptance.

1 INTRODUCTION

This research investigates the application of advanced deep learning algorithms, specifically YOLOv9, to enhance automated detection of Personal Protective Equipment (PPE) in hazardous work environments. Automated PPE detection aims to ensure worker safety by identifying whether individuals wear necessary gear, such as hard hats and face shields. Traditional methods rely on convolutional neural networks (CNNs) for object recognition but face limitations in accuracy and adaptability under variable conditions. Although prior research has shown promise, gaps remain in achieving reliable real-time PPE detection in complex settings.

---

* Place the footnote text for the author (if applicable) here.

Studies have explored various YOLO-based models for PPE detection, each highlighting advantages and limitations. For example, study [1] used YOLOv4 and YOLOv5 models trained with a dataset of 11,579 images of PPE classes across diverse lighting, angles, and distances in a laboratory setting. YOLOv5 showed the highest accuracy (mAP of 88.6%), although limited dataset diversity reduced the model's generalizability to real-world environments. Study [2] implemented SafeFac with YOLOv3 for assembly line monitoring. By combining YOLOv3 with RGB histogram filtering and cosine similarity-based post-processing, the model reached impressive metrics (precision 99.93% and recall 96.44%), demonstrating potential in controlled factory settings but requiring further tuning to avoid false negatives in diverse conditions.

In construction site monitoring, study [3] employed YOLOv5 in four configurations (YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x) to address the need for fast, accurate detection of PPE. With an 11,978-sample dataset, YOLOv5s achieved a detection rate of 110 frames per second, meeting real-time detection requirements. However, varied lighting and angle conditions affected the model's generalization. Similarly, study [4] utilized a vision-based system integrating OpenPose and YOLOv3 to monitor PPE usage at Fukushima's decommissioning site, achieving precision and recall of 97.64% and 93.11%, respectively. This system proved effective but was sensitive to occlusions and weather variability, limiting its robustness in diverse settings.

In [5], the authors compared eight YOLOv4-based models trained on the CHV dataset for construction site PPE detection. While YOLOv5x showed the highest mAP (86.55%) and YOLOv5s the fastest processing speed (52 FPS), helmet detection suffered on blurred images, underscoring the impact of clarity and visibility in real-world deployment.

Collectively, these studies reveal a primary gap in existing research: the need for models that maintain high accuracy across varying environmental conditions and equipment types. This thesis addresses this gap by leveraging YOLOv9, which incorporates improved adaptability and computational efficiency, aiming to enhance real-time PPE detection while minimizing false positives and negatives. By refining the model with a diverse, PPE-centered dataset and optimizing it for complex environments, this research contributes to safer industrial practices and future advances in automated safety compliance monitoring.

2 REVIEW OF RELATED LITERATURE

Ensuring workplace safety has driven significant research into real-time detection and monitoring of Personal Protective Equipment (PPE) compliance. The following studies highlight advanced approaches to PPE monitoring, addressing technical methodologies, challenges, and potential areas for improvement.

2.1 Pixel-Level Detection for PPE Usage

Yeo-Reum Lee et al. (2023) developed a model for pixel-level detection of PPE compliance, focusing on instance segmentation to identify whether individuals are properly wearing protective gear. Their system integrates YOLACT for object detection with MobileNetV3 as the lightweight backbone, combined with the DeepSORT algorithm to track PPE across video frames. The dataset includes a diverse set of images from surveillance cameras, smartphones, and public resources, annotated with masks and bounding boxes for various PPE classes. The model achieved 91.3% accuracy on mean Average Precision (mAP) metrics at different IoU thresholds, demonstrating effective real-time PPE status detection suitable for high-risk construction sites. This approach exemplifies the value of instance segmentation and object tracking in PPE compliance monitoring. [6]

2.2 Automated PPE Violation Detection in Manufacturing

Fasinu (2023) presented an automated PPE detection system tailored for manufacturing environments. Utilizing BrainFrame software with Intel's OpenVINO, this system enables real-time processing of video feeds to detect PPE compliance. The setup included over 3,000 images of PPE items like helmets, safety glasses, and N95 masks, ensuring robust identification across varied operational conditions, including worker posture and lighting. Fasinu's study underwent three phases: a survey to identify PPE compliance challenges, a pilot study at WVU's Lane Manufacturing Hub, and final lab testing under simulated conditions. This staged approach allowed for refinement of the model, emphasizing its potential for improving industrial safety compliance through timely and automated PPE detection. [7]

2.3 Bluetooth Low Energy for PPE Monitoring

Pisu et al. (2024) explored the use of Bluetooth Low Energy (BLE) and RSSI signals for PPE monitoring. Their model employs a Support Vector Machine (SVM) to assess PPE compliance based on RSSI patterns, specifically detecting correct PPE usage without relying on distance measurements. The architecture comprises BLE devices on PPE (e.g., helmets and shoes) and a post-processing algorithm to limit false positives. Testing achieved an accuracy of 88.1% and an F-measure of 90.2%, illustrating BLE's viability for continuous, non-intrusive monitoring in dynamic environments. The four-layer IoT system architecture—sensors, edge, fog, and cloud—facilitates efficient data processing and management, enhancing overall monitoring effectiveness. This innovative BLE approach supports real-time PPE detection while addressing scalability and accuracy challenges. [8]

2.4 Image Detection and Object Recognition with Smart Hard Hats

Habbal et al. (2019) developed a system using image detection and object recognition models for monitoring PPE use on construction sites. By employing template matching algorithms and face detection, the system identifies proper PPE usage, including helmets, goggles, and clothing. Real-time images captured by smart hard hat cameras are processed to detect PPE compliance, achieving high accuracy (95.1% for heads, 97.2% for bodies, 87.4% for hands, and 56.3% for shoes). The system generates alerts for safety violations and supports real-time compliance monitoring by site managers. The workflow includes detecting humans, capturing images, checking PPE, verifying registration, and generating alerts for non-compliance or unregistered individuals. [9]

2.5 PPE Compliance Detection with Hybrid AI Architecture

Balakreshnan et al. (2020) developed a PPE compliance system using Microsoft Azure's AI services and hybrid cloud-local processing. Using a TensorFlow-based object detection algorithm trained on both public and local images, the system achieved significant improvements in precision (155%), recall (79.1%), and mAP (233%) after retraining with locally collected data. The system integrates edge computing, where IoT-enabled cameras capture images and generate alerts for PPE violations, transmitting the data to an IoT Hub for further processing and visualization. The final alerts are displayed on a dashboard after processing. [10]

2.6 Hyperparameter Optimization in Deep Learning

Dahiya et al. (2023) developed a YOLOv5-based model for skin lesion detection using a dataset of monkeypox images. The model utilized several optimization techniques, including the SDG optimizer, Bayesian optimization, and Optuna's TPE method for hyperparameter tuning. It achieved 98.28% accuracy with 0.991 precision, 0.928 recall, and an mAP of 0.887. The system outperformed existing models in terms of accuracy. The model used Google Colab for implementation, and comparisons were made with models like DenseNet201, ResNet5, and SVM. [11]

3 METHODOLOGY

This section details a method for improving construction site safety through automated PPE detection using deep learning-based image recognition. To refine the approach to PPE detection, specifically helmets, an advanced feature extraction and association process has been implemented. This method ensures that detected helmets are accurately recognized and correctly associated with the construction workers wearing them.

Identifying Helmets with Precision: The model leverages the YOLOv9 architecture, trained to pinpoint specific helmet characteristics such as shape, color, and texture. This enables precise differentiation of helmets from other objects in varied environments, increasing detection accuracy for safety compliance.

Associating Helmets with Workers: Upon detecting a helmet, the model assesses its spatial relationship to individuals in the frame. By examining proximity and alignment, it ensures that each helmet is correctly linked to a person, avoiding misidentification with background elements or other objects.

Integration with Safety Monitoring: This process integrates into a comprehensive safety monitoring system used on construction sites. Designed to adapt and learn from ongoing data input, the system can handle variations in helmet types and environmental conditions, ensuring continuous relevance and effectiveness in safety enforcement.

3.1 Experimental Environment

The experimental environment was carefully selected to ensure the efficient development, training, and deployment of the YOLOv9-based PPE detection model. A combination of software tools, high-performance hardware, and specific peripheral devices facilitated the model’s demand for real-time image processing, data handling, and high-speed training operations. This setup integrated a robust object detection framework, essential libraries for deep learning, command-line tools for smooth automation, and a high-definition webcam for direct data capture. The arrangement provided a stable environment for effective experimentation and accurate, reproducible results.

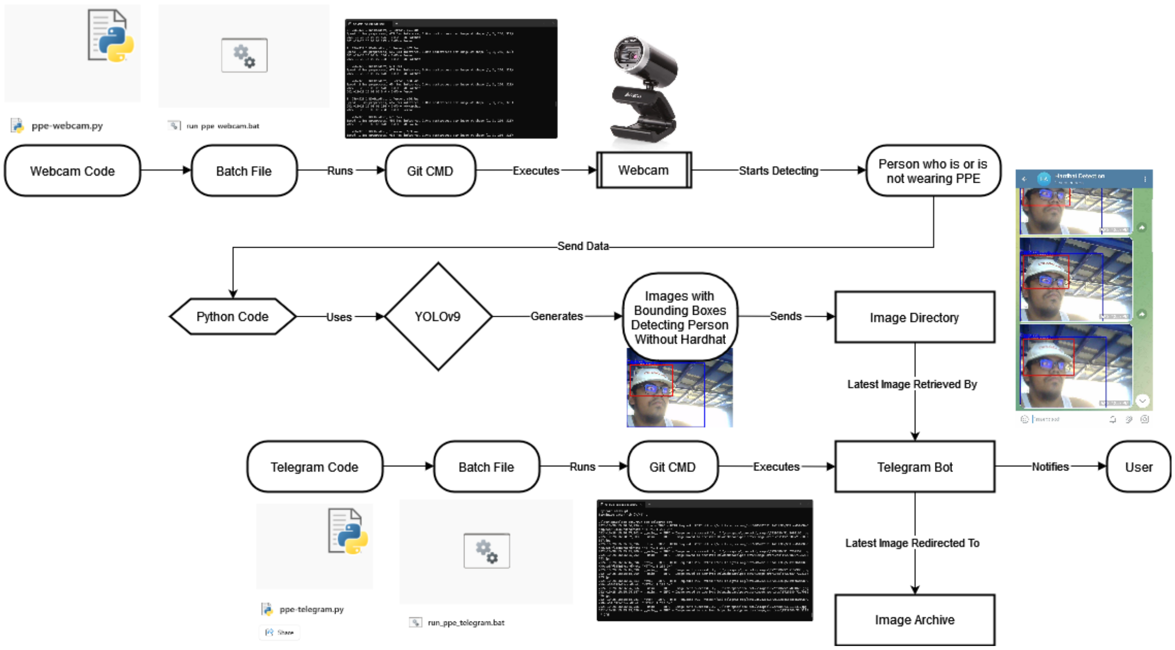

Figure 1: Model Framework for YOLOv9 PPE Detection System

This framework illustrates an automated detection and alert system designed to identify PPE compliance, specifically the presence or absence of a hardhat on individuals in real time. The process begins with a webcam feed analyzed through a Python script implementing the YOLOv9 model, which is executed via a batch file in Git CMD. As the script runs, it activates the webcam to scan the environment, applying bounding boxes to any detected persons and classifying their hardhat status. When a person is detected without a hardhat, the system immediately captures an image, saving it in a designated folder for further processing. A second Python script, also triggered by a batch file in Git CMD, retrieves the latest saved image and sends it as an alert notification to the user via Telegram. Once the image is successfully sent, it is automatically moved to an archive folder for record-keeping, ensuring that each detection is stored in an organized, timestamped manner for future review. This end-to-end process demonstrates an efficient workflow for PPE monitoring, combining YOLOv9's object detection with real-time alerts and systematic archiving of non-compliant events.

The YOLOv9 framework served as the primary object detection model, chosen for its accuracy and real-time efficiency in detecting hardhats on individuals within a construction environment. Python, along with libraries such as PyTorch for deep learning, OpenCV for image processing, and the Telegram API for sending alerts, was used as the main programming language. A high-resolution A4tech HD 1080p webcam provided clear video input, while Git CMD streamlined command-line operations for automating the detection, notification, and archiving processes. Google Colab enabled GPU acceleration, improving model performance and reducing training time, while local hardware managed data preprocessing before cloud-based training. Custom Python scripts were employed to handle data annotation, dataset updates, and archiving, ensuring consistency and efficiency across the workflow.

3.2 Conceptual Framework

The experimental environment was carefully selected to ensure the efficient development, training, and deployment of the YOLOv9-based PPE detection model. A combination of software tools, high-performance hardware, and specific peripheral devices facilitated the model's demand for real-time image processing, data handling, and high-speed training operations. This setup integrated a robust object detection framework, essential libraries for deep learning, command-line tools for smooth automation, and a high-definition webcam for direct data capture. The arrangement provided a stable environment for effective experimentation and accurate, reproducible results.

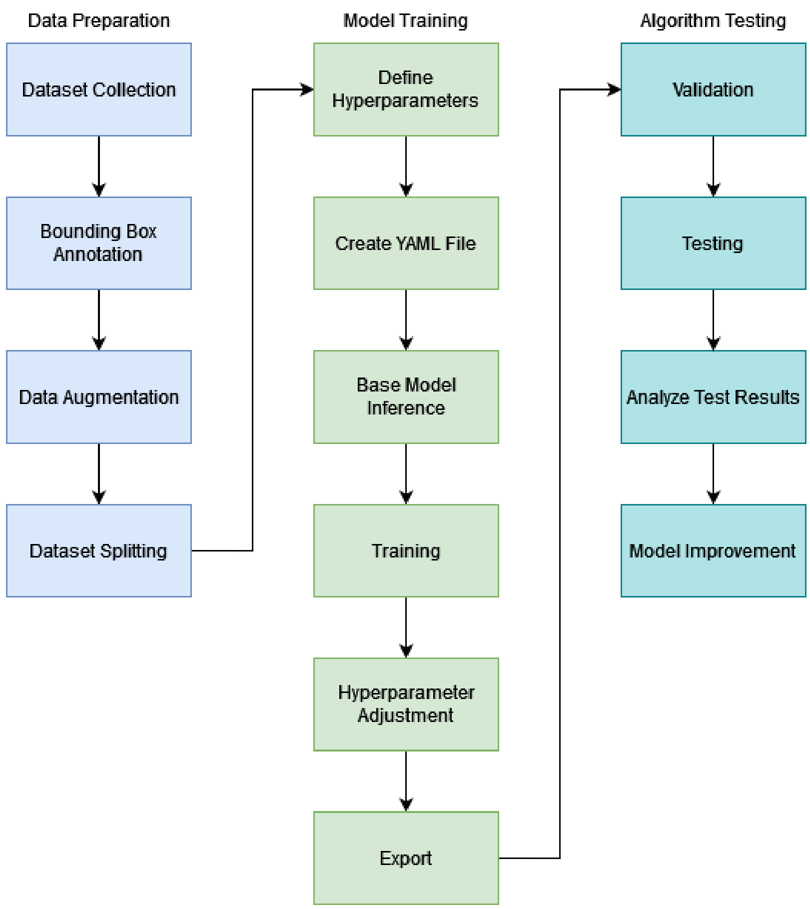

Figure 2: Conceptual Framework of YOLOv9 Finetuning for PPE Detection

This framework illustrates the conceptual framework of YOLOv9 finetuning for PPE detection, encompassing three stages: Data Preparation, Model Training, and Algorithm Testing. Data Preparation begins with dataset collection from various sources, including personal camera footage, Kaggle datasets, Roboflow, YouTube, and real-world video footage from a construction site in Carmona, Cavite. This data is then annotated by drawing bounding boxes around PPE items, ensuring that only relevant images and properly labeled data are used. Data augmentation techniques such as rotation, scaling, flipping, and color adjustments are applied to enhance dataset diversity. The dataset is then split into training, validation, and test sets to ensure robust model evaluation.

Model Training involves several steps, starting with defining hyperparameters like learning rate and batch size, followed by creating a YAML file for dataset configuration. Base model inference establishes initial performance, which is used to adjust hyperparameters iteratively. Training the model involves feeding annotated images into YOLOv9, updating weights through multiple epochs, and performing validation checks to monitor overfitting or underfitting. Once trained, the model is exported for use in real-world applications.

Testing the algorithm includes evaluating model performance on the validation and test sets, calculating metrics such as accuracy, precision, recall, and F1-score. Visual tools like confusion matrices, composite visualizations, and correlograms are used to assess accuracy, efficiency, and relationships between different metrics. Analyzing test results informs model improvement, with adjustments made to hyperparameters or training strategies to address any performance issues, such as false positives or instability.

3.3 Data Preparation

This study utilizes the "Construction Site Safety Image Dataset" from Roboflow (licensed under CC BY 4.0) [12]. This dataset is particularly valuable for deep learning model training due to its labeled images for "Hardhat," "NO-Hardhat," and "Person," enabling accurate assessment of PPE compliance.

Dataset Summary
Classes: 3
Annotations: YOLO format (.txt)
Metadata: Includes metadata.csv and count.csv for dataset statistics and train-val-test splits.
PPE Class Mapping: {0: 'Hardhat', 1: 'NO-Hardhat', 2: 'Person'}
Structure: Organizes images and labels for training, validation, and test in YOLO-compatible format.

Bounding box annotation is crucial to accurately label PPE items in collected images, ensuring ground truth consistency for model training. Using Roboflow, annotators will label hardhats and persons across varying angles, distances, and lighting conditions, which will improve the model’s robustness. Post-annotation, data will be cleaned to remove unused or incorrect labels, minimizing errors during training and evaluation.

To improve generalization, data augmentation will enhance the dataset’s diversity by applying transformations, including rotation, scaling, flipping, and color adjustments. These augmentations simulate various environmental conditions and will be implemented using Albumentations and YOLOv9’s built-in methods. This approach prevents overfitting, making the model more resilient to variations in real-world scenarios.

Dataset splitting is essential for preparing the data to train the YOLOv9 model effectively for PPE detection. This process involves dividing the dataset into three parts: the training, validation, and test sets. The training set, comprising the bulk of the data, will be used to teach the model to recognize features associated with PPE items like hardhats and persons. The validation set, a smaller separate subset, will serve to fine-tune hyperparameters and evaluate the model's performance during training, helping to prevent overfitting by allowing adjustments to factors such as learning rate and batch size. Finally, the test set, kept completely separate from training and validation, will provide an unbiased evaluation of the model's ability to generalize to new, unseen data. Using the test set only after training and validation will yield performance metrics—such as accuracy, precision, recall, and F1-score—that gauge the model's effectiveness in real-world conditions. This structured splitting process is crucial for developing a robust, reliable PPE detection model.

3.4 Modelling

This section describes the configuration and training of the YOLOv9 model for PPE detection. Initialized with pretrained weights from "yolov9e.pt," the model underwent training over 70 epochs to achieve a balance between learning depth and avoidance of overfitting. A batch size of 16 was chosen to optimize computational efficiency and training performance, allowing for effective parameter optimization.

During training, iterative adjustments to key configuration parameters (CFG) were made to enhance model accuracy. Hyperparameters, including learning rate, batch size, and epochs, were carefully monitored and adjusted based on loss values, precision, recall, and F1-score metrics. For example, if validation loss began increasing while training loss decreased, indicating overfitting, epochs were reduced, and the model architecture simplified. Conversely, if both losses remained high, additional epochs or complexity adjustments were applied to address underfitting.

The learning rate was tuned based on observed changes in training and validation losses. A smooth decrease in loss indicated a well-set learning rate, while erratic or slow changes required adjustments. Batch size was adapted based on GPU memory usage, avoiding out-of-memory errors while ensuring stable gradient estimates. To determine the appropriate number of epochs, convergence criteria were applied: if validation loss improved less than 1% over 10 epochs, early stopping was triggered.

Validation set metrics were used to track model performance. For overfitting, architecture complexity was reduced, while underfitting was addressed by increasing layers or neurons. Adjustments aimed to achieve an optimal balance between training depth and resource usage. By iteratively monitoring and fine-tuning these parameters, YOLOv9 was optimized for accurate and robust PPE detection, capable of performing well under varied real-world conditions.

3.5 Testing and Validation

The testing and validation process involves preparing annotated datasets, running model inference with non-maximum suppression, and matching predictions to ground truth using IoU. By computing precision, recall, F1 score, AP, and mAP, the model's performance in PPE detection is evaluated, enabling adjustments for optimal accuracy in diverse, real-world environments.

Performance metrics are essential for assessing the accuracy and efficiency of object detection models, specifically in evaluating their ability to identify and localize objects and manage false positives and false negatives. These metrics provide insights into the model's performance, guiding further refinements [13].

3.5.1 Precision and Recall

Precision measures the proportion of true positives among all positive predictions, indicating the model's capacity to avoid false positives. **Recall**, by contrast, assesses the proportion of true positives among actual positives, reflecting the model's ability to detect all instances of a class. These metrics rely on the confusion matrix, which includes True Positives (TP), False Positives (FP), False Negatives (FN), and True Negatives (TN) for detailed performance analysis:

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

3.5.2 F1 Score

The **F1 Score** is the harmonic mean of precision and recall, providing a balanced performance evaluation. It considers both false positives and false negatives, making it useful for understanding the model's overall accuracy:

$$F1 = \frac{\text{Precision } \times \text{Recall}}{\text{Precision } + \text{Recall}}$$

3.5.3 Intersection Over Union (IoU)

IoU quantifies the overlap between a predicted bounding box and the ground truth box, assessing object localization accuracy. It is computed as the area of intersection divided by the area of the union between the predicted and ground truth bounding boxes:

$$IoU = \frac{\text{Area}(B_p \cap B_g)}{\text{Area}(B_p \cup B_g)}$$

$$IoU = \frac{\text{Area of Intersection}}{\text{Area of Union}}$$

3.5.4 Average Precision (AP)

AP measures the area under the precision-recall curve, summarizing the model's precision and recall across different confidence thresholds. This provides a single score to represent performance for each class:

$$AP = \int_0^1 Precision \times Recall\, dr$$

3.5.5 Mean Average Precision (mAP)

mAP extends AP by averaging AP scores across all classes, offering a comprehensive performance metric for multi-class object detection:

$$mAP = \frac{1}{\text{N}} \sum_{i=1}^{N} AP_i$$

4 RESULTS AND DISCUSSION

The YOLOv9 model was trained and tested on a dataset containing images of construction site workers with "Hardhat," "NO-Hardhat," and "Person" labels. Through meticulous data preparation, including annotation and augmentation, the dataset was enhanced to support robust training. In the Google Colab environment, libraries such as ultralytics and wandb facilitated smooth integration, tracking, and monitoring throughout the training process.

In this study, the YOLOv9 model was optimized for PPE detection by training on a robust dataset labeled with "Hardhat," "NO-Hardhat," and "Person" classes. The CFG class details the model's configuration settings, while the Dataset Statistics provides an overview of the training, validation, and test set distributions. This data setup ensured comprehensive class representation across varied environmental conditions.

ALGORITHM 1: CFG Class

```
class CFG:
    DEBUG = False
    FRACTION = 0.05 if DEBUG else 1.0
    SEED = 88
    CLASSES = ['Hardhat', 'NO-Hardhat', 'Person']
    NUM_CLASSES_TO_TRAIN = len(CLASSES)
    EPOCHS = 3 if DEBUG else 70
    BATCH_SIZE = 16
    BASE_MODEL = 'yolov9e'
    BASE_MODEL_WEIGHTS = f'{BASE_MODEL}.pt'
    EXP_NAME = f'ppe_css_{EPOCHS}_epochs'
    OPTIMIZER = 'Adam'
    LR = 5e-5
    LR_FACTOR = 0.1
    WEIGHT_DECAY = 1e-4
    DROPOUT = 0.0
    PATIENCE = 20
    PROFILE = False
    LABEL_SMOOTHING = 0.1
    CUSTOM_DATASET_DIR = '/kaggle/input/construction-site-safety-image-dataset-roboflow/css-data/'
    OUTPUT_DIR = './'
```

Table 1: Dataset Statistics for Training, Validation, and Test Sets

	Mode	Hardhat	NO-Hardhat	Person	Data_Volume
0	train	1473	1440	2703	2712
1	valid	57	44	101	101
2	test	36	28	67	68

Training was conducted over 70 epochs using yolov9e.pt weights for initialization, with batch size and learning rate tuned to optimize memory usage and computational efficiency. Training and Validation Curves (for F1, Precision-Recall, Precision, and Recall) demonstrate effective learning, with loss metrics for box, class, and DFL showing steady decreases, indicating convergence and minimal overfitting.

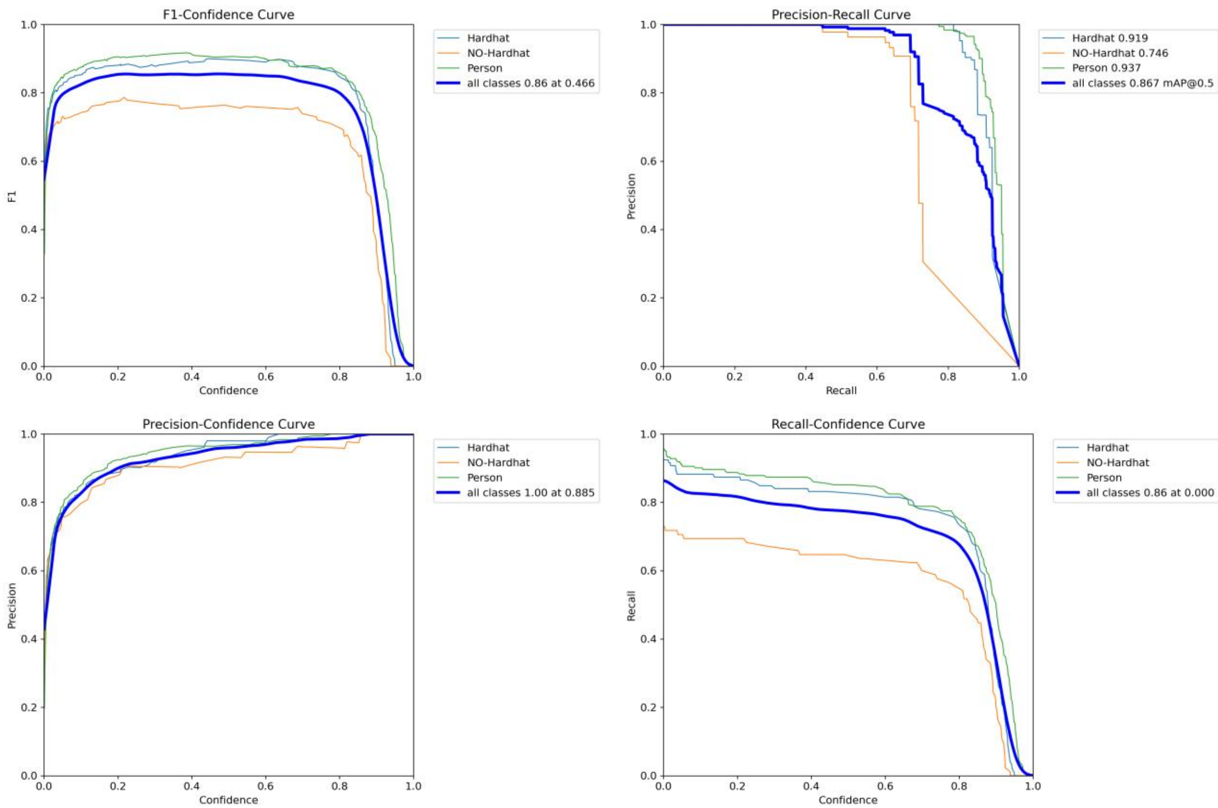

Figure 3: F1, PR, Precision, and Recall Confidence Curves

The Confusion Matrices (Raw and Normalized) reveal detection performance for each class. High precision in the Person class and solid recall for the NO-Hardhat class underscore the model's capacity to detect PPE compliance. Minor misclassifications in the Hardhat class were observed, particularly in low-light or cluttered scenes, likely due to visual similarities that caused false positives.

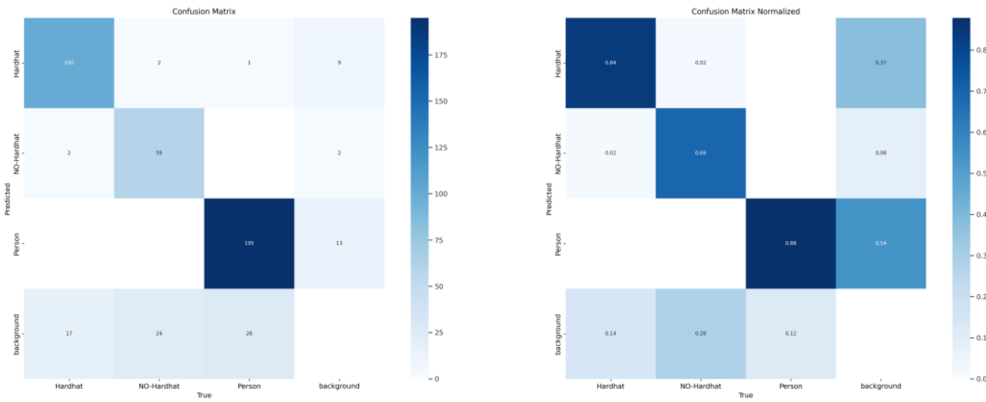

Figure 4: Confusion Matrices (Raw and Normalized)

To examine spatial relationships and bounding box characteristics, a Correlogram and Composite Visualization provide insights into object dimensions, label distributions, and positional heatmaps. The Composite Visualization also illustrates class distribution and detection density across the dataset, confirming that the model effectively identifies PPE items in diverse contexts.

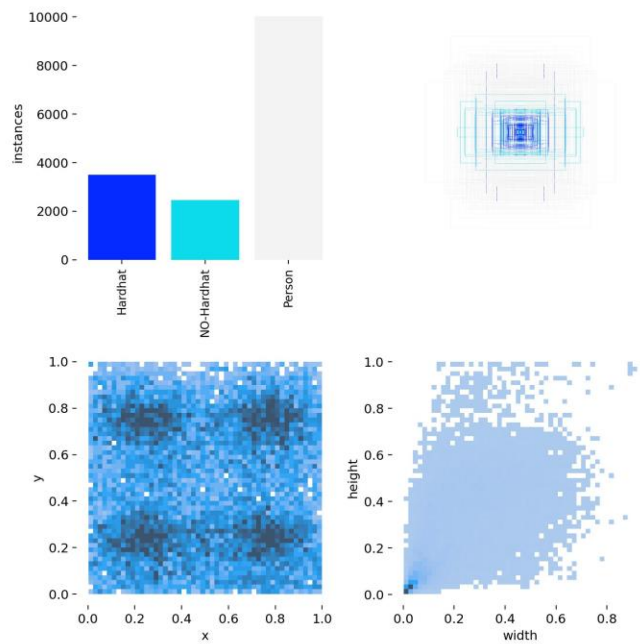

Figure 5: Composite Visualization of Class Distributions and Object Detection Metrics

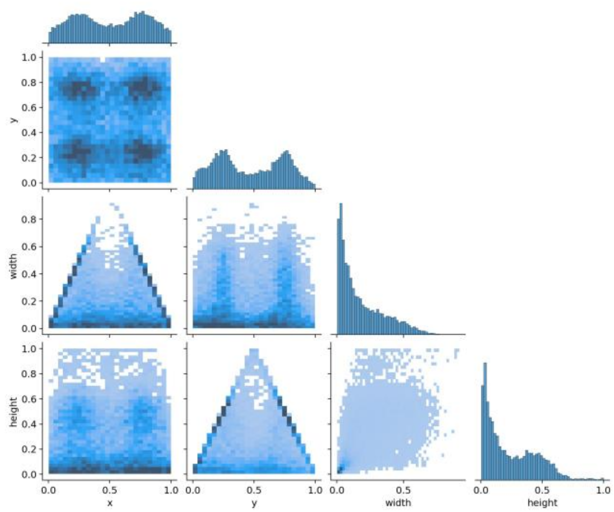

Figure 6: Distribution Analysis (Correlogram) of Object Detection Bounding Box Characteristics

The Training Dynamics figure highlights model performance across epochs, showing consistent improvements. As shown in the Table of Training Metrics Progression Over 70 Epochs, consistent improvements in precision, recall, and mAP indicate that hyperparameter adjustments enhanced model performance. Finally, Validation Results and Loss Graphs (Box, Cls, and DFL) validate the model's robustness, with metrics stabilizing across training epochs.

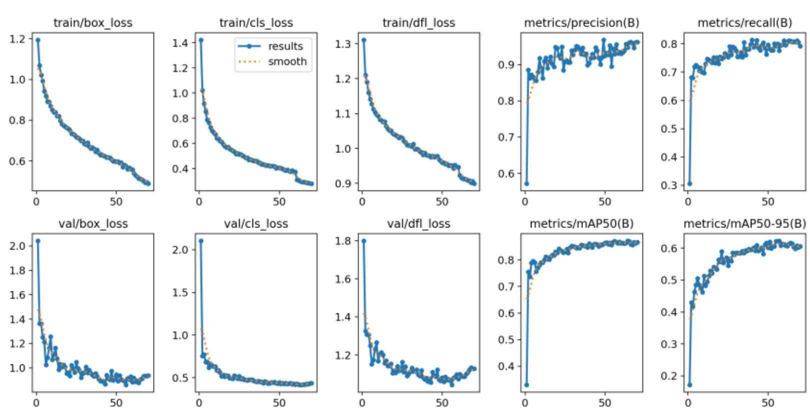

Figure 7: Training Dynamics of an Object Detection Model

Table 2: Progression of Object Detection Model Training Metrics Over 70 Epochs

epoch	train/box_loss	train/cls_loss	train/dfl_loss	metrics/precision(B)	metrics/recall(B)	metrics/mAP50(B)	metrics/mAP50-95(B)	val/box_loss	val/cls_loss	val/dfl_loss	lr/pg0	lr/pg1	lr/pg2
1	1.1944	1.424	1.3113	0.57135	0.30581	0.33031	0.17216	2.0421	2.1043	1.7993	0.066879	1.66E-05	1.66E-05
2	1.0686	1.0237	1.21	0.88537	0.68083	0.75511	0.42973	1.3637	0.75598	1.3248	0.033562	3.28E-05	3.28E-05
3	1.0211	0.91509	1.1901	0.86254	0.68022	0.73699	0.41661	1.3622	0.77212	1.3073	0.000245	4.86E-05	4.86E-05
4	0.99279	0.85172	1.16	0.87213	0.71901	0.78872	0.46306	1.2513	0.68167	1.3067	4.81E-05	4.81E-05	4.81E-05
1	1.1944	1.424	1.3113	0.57135	0.30581	0.33031	0.17216	2.0421	2.1043	1.7993	0.066879	1.66E-05	1.66E-05
...	...	...	...	...	...	...	...	...	...	...	...	...	...
66	0.50696	0.29011	0.90839	0.96215	0.80806	0.85917	0.604	0.9102	0.42077	1.1103	8.21E-06	8.21E-06	8.21E-06
67	0.50316	0.28723	0.90873	0.95866	0.8065	0.85629	0.5978	0.93036	0.42539	1.1223	7.57E-06	7.57E-06	7.57E-06
68	0.49474	0.28523	0.90264	0.94584	0.81184	0.86641	0.60744	0.93271	0.42407	1.1335	6.93E-06	6.93E-06	6.93E-06
69	0.49523	0.28362	0.90604	0.96108	0.80459	0.86247	0.60266	0.9335	0.4305	1.1285	6.29E-06	6.29E-06	6.29E-06
70	0.48858	0.27945	0.89926	0.96264	0.7922	0.86719	0.60555	0.93658	0.43394	1.1291	5.64E-06	5.64E-06	5.64E-06

Figure 8: Box, Cls, and DFL Loss

Figure 9: Validation Results

Each image shows the YOLOv9 model’s recognition of objects such as "Person," "Hardhat," and "NO-Hardhat," with labeled bounding boxes color-coded by class. These bounding boxes highlight the model's detection performance, capturing both successful classifications and instances where the model misclassifies or misses objects, especially in "NO-Hardhat" scenarios. Accurate boxes around hardhats and persons indicate the model’s effectiveness, while missed or incorrect labels suggest areas for improvement.

Key Observations: The model reliably detects "Hardhat," "NO-Hardhat," and "Person," demonstrating robustness across various contexts and lighting conditions. However, false positives occasionally occur, such as labeling “NO-Hardhat” on a hardhat-wearing individual, typically due to challenging visual conditions:

Lighting Issues: Shadows or reflections often interfere, causing the model to mistake hardhats for non-compliance.

Angles and Obstructions: Partial views, such as side or back angles, hinder accurate detection.

Similar Objects: Non-standard headgear resembling hardhats in shape or color can mislead the model.

Improvement Recommendations: Enhancing model accuracy may involve expanding the dataset with diverse lighting, angles, and headgear, adjusting detection thresholds, and fine-tuning with additional annotated samples. Post-processing techniques, such as multi-frame validation, could further reduce false positives, making the model more effective in real-world applications.

5 SUMMARY, CONCLUSION, AND RECCOMENDATION

This study developed and evaluated a YOLOv9-based deep learning model for detecting Personal Protective Equipment (PPE) in hazardous work environments. The research involved data preparation, model training, and testing on a curated dataset specifically designed for PPE detection. Performance metrics, including precision, recall, and mean Average Precision (mAP), revealed the model's effectiveness in accurately identifying and classifying PPE, with notable improvements in reducing false positives and negatives. The results indicate enhanced accuracy and generalizability across varied conditions, demonstrating the potential for this model to support workplace safety.

The conclusions drawn from this study confirm that the YOLOv9 model, with proper training and fine-tuning, is a robust tool for automated PPE detection. Its accuracy in identifying hardhats, distinguishing persons, and minimizing background interference suggests strong applicability in real-world settings where safety compliance is essential. While minor misclassifications between similar classes, such as "Hardhat" and "Person," were observed, the model's overall performance indicates that it can be a valuable asset in enhancing safety protocols. This research highlights the importance of data diversity and augmentation for model generalizability, especially when dealing with complex environments.

To further improve model effectiveness, future work should prioritize expanding data diversity to include a broader range of PPE types and environmental conditions, enhancing adaptability through transfer learning, and optimizing for real-time deployment. Integrating the model with edge computing would support faster processing, essential for live safety monitoring. Additionally, implementing long-term feedback loops for continuous learning could enhance model adaptability as environments and PPE standards evolve. Broadening detection capabilities to encompass other PPE types, such as vests, gloves, and boots, would make the model even more comprehensive for workplace safety applications.

REFERENCES

[1] Nugraha, K. O.P.P., and Rifai, A. P.. 2023. "Convolutional Neural Network for Identification of Personal Protective Equipment Usage Compliance in Manufacturing Laboratory." Jurnal Ilmiah Teknik Industri 22, 1 (June 2023), 11-24. DOI: https://doi.org/10.23917/jiti.v22i1.21826

[2] Ahn, J., Park, J., Lee, S. S., Lee, K-H., Do, H., and Ko, J. 2023. "SafeFac: Video-based smart safety monitoring for preventing industrial work accidents." Expert Systems with Applications 215, C (April 2023), 119397. DOI: https://doi.org/10.1016/j.eswa.2022.119397

[3] Ngoc-Thoan, N., Thanh Bui, D-Q., Tran, C. N. N., and Tran, D-H. 2024. "Improved detection network model based on YOLOv5 for warning safety in construction sites." International Journal of Construction Management, 24(9) (2023), 1007-1017. DOI: https://doi.org/10.1080/15623599.2023.2171836

[4] Chen, S. and Demachi, K. 2020. A Vision-Based Approach for Ensuring Proper Use of Personal Protective Equipment (PPE) in Decommissioning of Fukushima Daiichi Nuclear Power Station. Appl. Sci., 10, 5129. DOI:10.3390/app10155129.

[5] Wang, Z., Wu, Y., Yang, L., Thirunavukarasu, A., Evison, C., and Zhao, Y. 2021. Fast Personal Protective Equipment Detection for Real Construction Sites Using Deep Learning Approaches. Sensors, 21, 3478. DOI: https://doi.org/10.3390/s21103478.

[6] Lee, Y-R., Jung, S-H., Kang, K-S., Ryu, H-C., and Ryu, H-G. 2023. "Deep learning-based framework for monitoring wearing personal protective equipment on construction sites." Journal of Computational Design and Engineering, 10, 2 (March 2023), 905–917. DOI: https://doi.org/10.1093/jcde/qwad019

[7] Fasinu, J. O. 2023. "Improving the Health and Safety of Manufacturing Workers by Detecting and Addressing Personal Protective Equipment (PPE) Violations in Real-Time with the Use of Automated PPE Detection Technology." Graduate Theses, Dissertations, and Problem Reports. West Virginia University. https://researchrepository.wvu.edu/etd/12222/.

[8] Pisu, A., Elia, N., Pompianu, L., Barchi, F., Acquaviva, A., and Carta, S., 2024. Enhancing workplace safety: A flexible approach for personal protective equipment monitoring. Expert Systems With Applications. 238: 122285. https://doi.org/10.1016/j.eswa.2023.122285.

[9] Habbal, F., Habbal, F., Al Nuaimi, A., Al Shimmari, A., Safi, A., and AbuShqair, T. 2019. PPEs Compliance Technology to Legalize the Automated Monitoring of Safety Standards. In Proceedings of the 36th International Symposium on Automation and Robotics in Construction (ISARC 2019). DOI: 10.22260/ISARC2019/0113.

[10] Balamurugan, B., Richards, G., Nanda, G., Mao, H., Athinarayanan, R., Zaccaria, J. 2020. PPE Compliance Detection using Artificial Intelligence in Learning Factories. In Procedia Manufacturing, 45 (2020), 277-282. Elsevier Ltd. DOI: 10.1016/j.promfg.2020.04.017.

[11] Dahiya, N., Sharma, Y. K., Rani, U., Hussain, S., Nabilal, K. V., Mohan, A. & Nuristani, N. 2020. Hyper-parameter tuned deep learning approach for effective human monkeypox disease detection. Sci Rep 13, 15930 (2023). https://doi.org/10.1038/s41598-023-43236-1.

[12] Roboflow Universe Projects. 2023. "Construction Site Safety Dataset. Roboflow Universe." Roboflow. https://universe.roboflow.com/roboflow-universe-projects/construction-site-safety.

[13] Jocher, G., Munawar, M. R., and Vina, A.. 2024. "Performance Metrics Deep Dive." Ultralytics YOLOv8 Docs. https://docs.ultralytics.com/guides/yolo-performance-metrics/.