iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** VII **Month of publication:** July 2025

DOI: https://doi.org/10.22214/ijraset.2025.73462

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue VII July 2025- Available at www.ijraset.com

YOLO-M-Based Model for Detecting Safety Helmet Wearing At Construction Sites

Ramavath Srinivas[1], G Praveen Babu[2]
[1](Post Graduate Student, M. Tech (Data Sciences) Department of Information Technology, Jawaharlal Nehru Technological University Hyderabad
[2](Associate Professor, Department of Information Technology, Jawaharlal Nehru Technological University Hyderabad

Abstract: ***Ensuring safety on construction sites continues to be a major challenge, highlighting the need for intelligent and automated monitoring systems. This research presents a smart helmet detection framework based on advanced computer vision techniques, designed to enforce safety measures in real-time. The paper systematically evaluates multiple cutting-edge object detection models—such as YOLOv5s, YOLOv5-M, YOLOv3, YOLOv4, YOLOv8, YOLOv5 integrated with GhostCNN, SSD, RetinaNet, and Faster R-CNN. These models are assessed based on detection accuracy, processing speed, and hardware efficiency to determine their viability for safety enforcement tasks. The system primarily aims to safeguard construction personnel while also assisting site supervisors by enhancing resource management and surveillance. Experimental results indicate that the YOLOv5-GhostCNN architecture achieves a remarkable performance, attaining a mean Average Precision (mAP) exceeding 97%, highlighting its capability in critical safety applications. This work advances the goal of safer construction environments by promoting effective use of AI-based safety monitoring.***

Keywords: ***Smart surveillance, safety enforcement, object detection, YOLO variants, helmet compliance, feature fusion, attention-based models.***

I. INTRODUCTION

[1] In recent years, the emergence of intelligent technologies combined with deep learning algorithms has transformed several industries, significantly enhancing operational efficiency and safety protocols. Sectors like transportation and retail have widely adopted innovations such as facial recognition and license plate identification to streamline processes and strengthen security. However, the construction industry presents a distinct set of challenges due to its unpredictable and high-risk environment, especially regarding incidents like falling objects. In this context, safety helmets are essential protective gear, critical for minimizing head injuries and preserving workers' lives.

[2] Construction sites are known for their hazardous conditions, where falling tools or debris pose a major threat. Helmets function as a key safety measure, absorbing impact and preventing serious injuries or fatalities. Therefore, ensuring consistent usage of safety helmets on-site is vital to maintaining occupational safety. Historically, helmet compliance was monitored manually through site personnel, a method prone to inefficiencies and human error. Due to the vast and continuously changing work zones, this approach often failed to provide reliable enforcement and consumed valuable manpower resources.

[3] With the rise of deep learning and computer vision, a shift has occurred in how safety compliance is maintained on-site. These technologies allow for real-time, automated monitoring of workers, detecting whether helmets are being properly worn. Early systems using YOLO (You Only Look Once) detection frameworks showed strong real-time performance but often lacked precision, limiting practical implementation. Researchers responded by refining YOLO-based models—reducing model parameters, enhancing classification accuracy, and introducing new loss functions like IoU and GIoU to improve detection localization.

[4] Efforts to increase speed and deployability led to the development of lightweight models such as MobileNet-based YOLO variants, which lower computational requirements while maintaining acceptable detection accuracy. These advancements make the systems more feasible for real-time use in construction scenarios, especially when hardware capabilities are limited. Recent developments have introduced powerful techniques like Efficient Channel Attention (ECA) to boost feature discrimination and model performance. Furthermore, model pruning and sparse training methods have been adopted to streamline the models even further, enabling faster and more efficient execution.

[5] Looking into the future, researchers aim to incorporate cutting-edge approaches such as reinforcement learning and self-supervised learning, which can further enhance adaptability and intelligence in detection systems. Additionally, building robust and diverse datasets representing various construction environments and helmet types is essential for developing more generalized models.

©IJRASET: All Rights are Reserved

[6] In summary, the integration of deep learning for safety helmet detection has become a game-changer in improving workplace safety on construction sites. These intelligent systems ensure real-time surveillance, helping enforce safety protocols effectively and minimizing the risk of injury from falling objects. While earlier models faced limitations in accuracy, ongoing research continues to bridge these gaps. Through innovation and collaborative efforts between developers, industry professionals, and policy regulators, helmet detection systems are set to become a cornerstone of safe construction practices.

II. LITERATURE REVIEW

The construction sector is recognized as a high-risk environment, with workers often exposed to hazards such as falling debris and equipment. Safety helmets serve as a critical line of defense against head trauma in such conditions. Historically, enforcement of helmet-wearing policies relied on manual observation by site supervisors. However, this method has proven inefficient, inconsistent, and labor-intensive. The emergence of deep learning and computer vision has paved the way for automated helmet detection systems, greatly transforming safety enforcement practices on-site.

Li et al. [1] examined the ability of industrial safety helmets to resist impact under varying conditions. Their work contributed to a deeper understanding of helmet performance and emphasized the need for reliable safety gear as a foundation for automated detection systems.

Wang et al. [2] offered an extensive survey on algorithms used for detecting helmet usage in intelligent construction environments. The study explored multiple detection strategies, focusing particularly on deep learning-based models. This comprehensive review outlines the current advancements, limitations, and opportunities in the field of helmet detection.

Jun et al. [3] developed a real-time helmet detection system using the YOLO (You Only Look Once) framework. Their method showcased the high-speed processing capability of YOLO architectures, but also noted that such speed often comes at the cost of reduced detection accuracy—a common trade-off in real-time systems.

Wen et al. [4] enhanced the YOLOv3 architecture to improve the precision of helmet detection. Their modifications preserved the real-time benefits of YOLO while boosting accuracy, offering a more balanced solution for on-site implementation.

Ming et al. [5] proposed an optimized version of YOLOv2, focusing on minimizing computational demands while retaining detection performance. Their system proved effective for real-time usage, especially in fast-paced construction settings where speed is crucial.

Zhao et al. [6] introduced a lightweight variant, YOLO-S, designed specifically for environments with limited computational resources. By streamlining the model structure and minimizing parameter count, YOLO-S achieved reliable detection with significantly reduced hardware requirements, making it practical for deployment on edge devices.

Ding et al. [7] presented a refined YOLOX-based model that incorporated advanced feature extraction and customized loss functions. Their approach succeeded in enhancing both detection accuracy and operational speed, making it suitable for real-time monitoring in active construction zones.

III. METHODOLOGY

The proposed study is centered on improving safety helmet detection at construction sites by designing and evaluating advanced object detection frameworks. At the core of this work is YOLO-M (YOLO Mini), a compact and efficient variant derived from the YOLOv5s architecture [19], tailored to deliver high accuracy with reduced computational demand in crowded and dynamic construction environments.

To benchmark the performance of YOLO-M, a detailed comparative study will be carried out against well-known detection models such as SSD, RetinaNet, Faster R-CNN [14], YOLOv3 [4], and YOLOv4 [8]. This evaluation will help identify improvements in detection accuracy, speed, and overall robustness.

In addition to YOLO-M, enhanced versions of YOLOv5 will also be considered, including YOLOv5 integrated with GhostCNN, YOLOv8, and YOLOv5X6. These models will be analyzed for their ability to further improve detection outcomes. The comparative analysis will highlight the advantages and trade-offs among various architectures in the context of real-time helmet detection.

Moreover, the system will be implemented using the Flask web framework, paired with an SQLite database to support features such as user registration and login. This will allow for an interactive platform where detection outputs can be visualized and analyzed by users, enabling a comprehensive evaluation of both the technical performance and the usability of the proposed solution in realistic deployment scenarios.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue VII July 2025- Available at www.ijraset.com

A. System Architecture

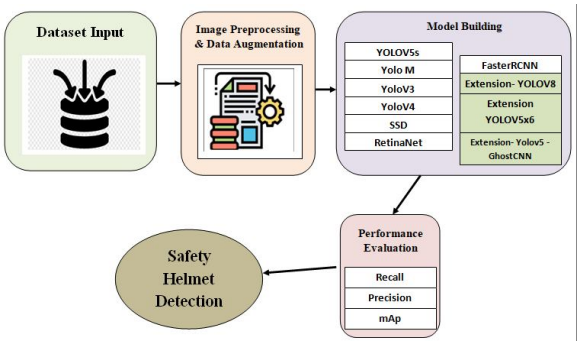


Fig 1 Proposed Architecture

The architecture of the proposed system initiates with the input of a labeled dataset, followed by preprocessing and data augmentation to improve model generalization and robustness. A range of object detection algorithms are developed and evaluated, including YOLO-M, SSD, RetinaNet, Faster R-CNN [14], YOLOv3 [4], and YOLOv4 [8], alongside enhanced YOLOv5 variants such as YOLOv5-GhostCNN [18], YOLOv8, and YOLOv5X6. To measure the effectiveness of each model, key evaluation metrics like precision, recall, and mean average precision (mAP) are employed. The model that achieves the best performance is chosen as the optimal detection solution for identifying safety helmets, ensuring reliability and precision in real-life construction site conditions.

B. Dataset Collection

For this project, the dataset was prepared using annotated images labeled through the Labelbox platform, with annotations converted into YOLOv5 PyTorch format using Roboflow. These images are sourced from actual construction sites and depict varied scenarios where safety helmets are either worn or absent. Each image includes detailed annotations highlighting the helmet's position using bounding boxes and class labels.

The dataset is designed to reflect diverse environmental conditions such as different lighting levels, weather situations, and camera angles. This ensures that the trained models can generalize well across various real-world conditions commonly encountered on construction sites. The annotations are provided in YOLO format, marking the exact location and class of helmets to support precise model training.

To facilitate structured training, the dataset is divided into training, validation, and testing sets. This distribution ensures effective learning, performance tuning, and unbiased evaluation of the detection models. The curated dataset forms the core of the detection framework, offering comprehensive inputs for building a robust safety helmet detection system.

C. Image Processing

1) *Blob Conversion:* The initial step involves reading the input image and transforming it into a blob—a format suitable for processing by deep learning models. This includes resizing the image to the required dimensions, scaling the pixel intensities, and applying normalization or mean subtraction as needed.
2) *Class Definition:* Next, the class of interest—in this case, "safety helmet"—is defined so that the model knows what object to detect during inference.
3) *Bounding Box Declaration:* Using annotation files, bounding boxes that indicate helmet locations within the image are retrieved. These coordinates identify the specific areas in the image where the helmets are situated.
4) *Conversion to NumPy Arrays:* After extracting blob data and bounding boxes, these elements are converted into NumPy arrays. NumPy provides efficient structures and operations for handling image data during preprocessing and model training.
5) *Loading the Pre-Trained Model:* The model is loaded using configuration and weight files that define the neural network's architecture and learned parameters. OpenCV's cv2.dnn.readNet() function is typically used for this task, enabling easy model integration.
6) *Output Layer Extraction:* After loading the network, its output layers—responsible for predictions such as object locations and confidence scores—are identified. These layers are essential for accessing and interpreting model output during inference.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

7) *Synchronizing Annotations:* The annotated image files and corresponding label data are loaded together, allowing the system to align image pixels with ground truth values for training and evaluation.
8) *Color Space Conversion (BGR to RGB):* If the input image is in BGR format (as is common with OpenCV), it is converted to RGB format to ensure compatibility with deep learning frameworks that expect RGB inputs.
9) *Mask Creation:* Based on the bounding boxes, masks are generated to highlight helmet regions. These masks focus model attention during training, improving learning efficiency.
10) *Image-Resizing:* Finally, images are resized to match the input shape expected by the neural network model. This step ensures consistency across all inputs, allowing the model to perform reliably during both training and testing.

D. Data Augmentation Steps

1) *Random Transformations:* To enrich the training dataset and make the model more robust, random transformations are applied to input images. These transformations include operations such as horizontal or vertical flipping, brightness variation, and scaling. This process helps simulate the visual diversity seen in real-world construction environments.
2) *Image Rotation:* Introducing random rotations allows the model to learn from various object orientations. This improves the algorithm's ability to detect safety helmets from multiple angles and in different positions, enhancing generalization.
3) *Affine Transformations:* Transformations such as scaling, translation, and rotation (affine transformations) mimic changes in camera perspective and object positioning. These augmentations increase the variability of training samples, which helps the model adapt to different visual scenarios during real-time deployment.

By implementing these augmentation strategies, the dataset is enriched with diverse variations, enabling the model to better handle practical conditions in construction sites and improving its detection reliability.

E. Algorithms

1) YOLOv5s: YOLOv5s is a fast and lightweight object detection model that segments an image into a grid and predicts bounding boxes along with class probabilities for each grid cell. The implementation involves loading a pre-trained model, resizing the input image to match the model's expected dimensions, and performing a forward pass to generate predictions. These predictions are then filtered using non-maximum suppression (NMS) to remove overlapping boxes and retain only the most confident detections for visualization or further use.
2) YOLO-M: YOLO-M is a custom, lightweight version of YOLOv5s, specifically designed for detecting safety helmets. It integrates MobileNetV3 as its backbone to reduce model size and computation time. The model also uses BiCAM (Bidirectional Channel Attention Module) for refined feature extraction and Res-FPN (Residual Feature Pyramid Network) for multi-scale detection. The workflow includes loading the YOLO-M model, preprocessing the image, obtaining predictions, and applying NMS to return accurate and efficient results.
3) YOLOv4: YOLOv4 offers a strong balance between speed and detection accuracy. The process includes loading a trained YOLOv4 model, resizing and normalizing the input image, and performing inference. After the forward pass, NMS is applied to filter out redundant or low-confidence detections, producing precise object localization outputs.
4) YOLOv3: YOLOv3 introduced the use of anchor boxes and multi-scale predictions to improve detection. The implementation starts with model loading and input image preprocessing, followed by inference. Post-processing involves applying NMS to clean the detection results before visualization or analysis.
5) YOLOv5-GhostCNN: This version incorporates the GhostNet backbone, designed to reduce computation while retaining feature richness. After loading the GhostCNN-enhanced YOLOv5 model, the image undergoes preprocessing and inference. Predictions are refined using NMS, resulting in a compact yet powerful helmet detection solution suitable for edge devices.
6) SSD (Single Shot MultiBox Detector): SSD uses default bounding boxes of different aspect ratios to detect objects in images. After loading the SSD model, the image is preprocessed and fed through the network. The detections are then refined using NMS to eliminate overlapping boxes, ensuring accurate and clean output.
7) RetinaNet: RetinaNet addresses class imbalance by introducing a focal loss function. The detection pipeline involves loading the pre-trained model, resizing the image, performing a forward pass, and refining the predictions using NMS to focus on the most relevant bounding boxes.
8) Faster R-CNN: Faster R-CNN follows a two-stage process. First, a Region Proposal Network (RPN) identifies potential object regions. These proposals are then classified in the second stage. After the model is loaded and the image is preprocessed, the RPN extracts candidate regions, which are refined through the classifier and finalized using NMS for output accuracy.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

9) YOLOv8: YOLOv8 introduces modern features like anchor-free detection, decoupled head, mosaic augmentation, and an upgraded loss function. After loading the model, the input image is resized and normalized. The model predicts objects without relying on predefined anchors, and NMS ensures only the most accurate detections are retained.

10) YOLOv5X6: YOLOv5X6 is a larger and deeper variant of the YOLO family, offering high detection precision at the cost of greater computational resources. The process involves loading the trained model, preprocessing the image, conducting inference, and applying NMS to retrieve accurate results, making it ideal for scenarios where hardware capacity is not a constraint.

IV. EXPERIMENTAL RESULTS

1) Precision: Precision evaluates the fraction of correctly classified instances or samples among the ones classified as positives. Thus, the formula to calculate the precision is given by:

$$Precision = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} (1)$$

2) Recall: Recall is a metric in machine learning that measures the ability of a model to identify all relevant instances of a particular class. It is the ratio of correctly predicted positive observations to the total actual positives, providing insights into a model's completeness in capturing instances of a given class.

$$Recall = \frac{TP}{TP + FN} (2)$$

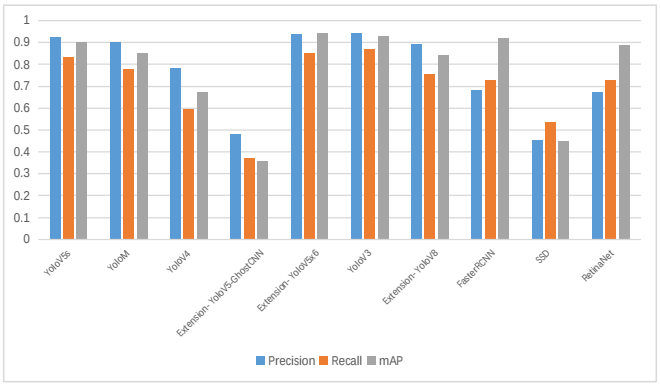
3) mAP: Mean Average Precision (MAP) is a ranking quality metric. It considers the number of relevant recommendations and their position in the list. MAP at K is calculated as an arithmetic mean of the Average Precision (AP) at K across all users or queries.

$$mAP = \frac{1}{n} \sum_{k=1}^{k=n} AP_k (3)$$

Table (1) evaluate the performance metrics—precision, recall, map, for each algorithm. Across all metrics, the YoloV3 consistently outperforms all other algorithms. The tables also offer a comparative analysis of the metrics for the other algorithms.

ML Model	Precision	Recall	mAP
YoloV5s	0.926	0.834	0.902
YoloM	0.904	0.780	0.853
YoloV4	0.785	0.596	0.675
Extension- YoloV5-GhostCNN	0.483	0.372	0.355
Extension- YoloV5x6	0.938	0.853	0.944
YoloV3	0.944	0.870	0.931
Extension- YoloV8	0.896	0.753	0.845
FasterRCNN	0.680	0.728	0.920
SSD	0.452	0.534	0.450
RetinaNet	0.675	0.728	0.890

Table.1 Performance Evaluation Table



Graph.1 Comparison Graph for Performance Evaluation Table.

Precision is represented in blue, recall in orange, map in gray and ***Graph (1)***. In comparison to the other models, the YoloV3 shows superior performance across all metrics, achieving the highest values. The graphs above visually illustrate these findings.

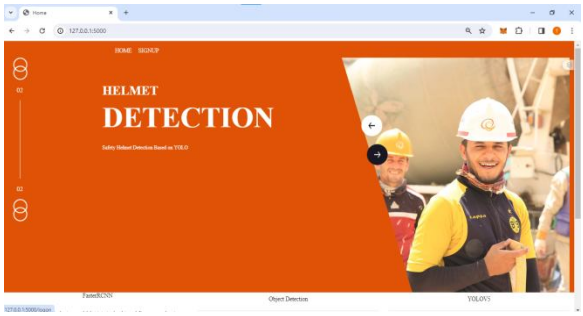


Fig 2 Home Page

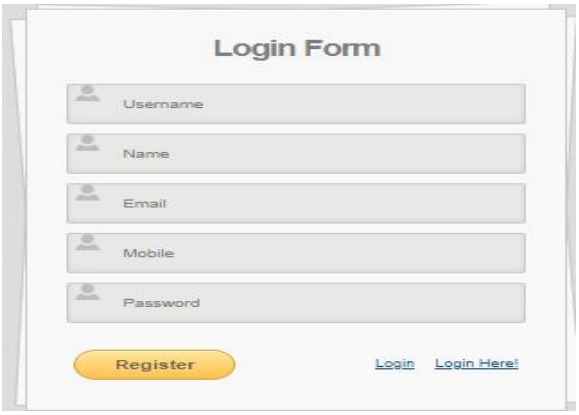


Fig 3 Registration Page

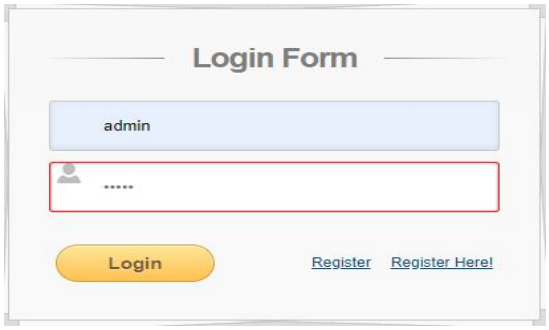


Fig 4 Login Page

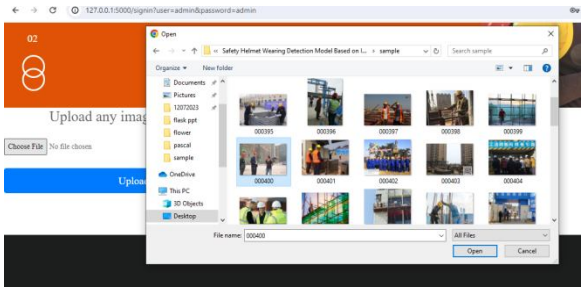


Fig 5 Upload Input Image



Fig 6 Final Outcome

V. CONCLUSION

In conclusion, the development of an automated safety helmet detection system represents a significant advancement in workplace safety within the construction industry. Through the utilization of computer vision technologies and cutting-edge algorithms such as YOLO variants, customized YOLO-M model, SSD, RetinaNet, and FasterRCNN [14], the project has effectively addressed the critical need for real-time monitoring of safety helmet compliance. The extension to explore additional algorithms like YOLOv5x6 [18] and YOLOv8 further enhances the system's robustness and accuracy. By integrating Flask with user authentication, the project ensures a user-friendly interface for testing and validation, facilitating practical deployment. Ultimately, these outcomes contribute tangibly to the construction industry by automating safety monitoring processes, aiding site managers and workers in maintaining a safer working environment.

VI. FUTURE SCOPE

Looking ahead, the future scope for safety monitoring systems in the construction industry is promising. Further refinement of detection algorithms and architectures, such as exploring advanced variants like YOLOv5X6, holds the potential to enhance accuracy and efficiency in safety helmet detection, thus improving overall workplace safety.

Incorporating emerging technologies like edge computing and real-time analytics presents an opportunity to enable on-device processing, facilitating instant detection and response in dynamic environments. This advancement could significantly enhance worker safety by providing timely alerts and interventions.

Expanding the scope beyond safety helmets to detect multiple safety gear items or potential hazards in construction sites would promote comprehensive safety measures, further mitigating risks and ensuring a safer work environment.

Integration with Internet of Things (IoT) devices offers seamless monitoring and management of safety protocols. This integration enables proactive safety measures and automated alerts in case of non-compliance, ultimately enhancing overall safety management in construction sites. By embracing these advancements, safety monitoring systems can continue to evolve, ensuring continuous improvement in workplace safety standards.

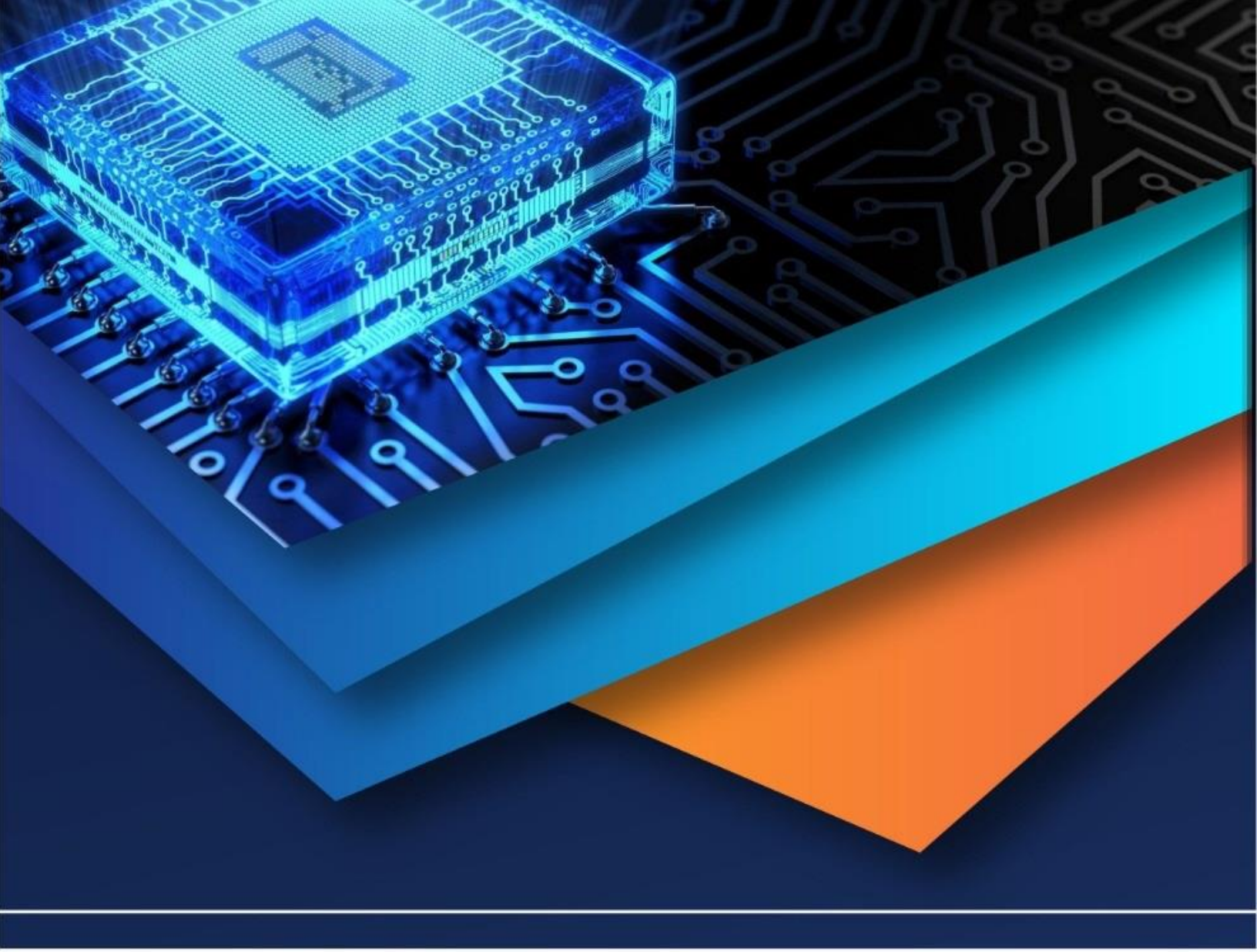
©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

REFERENCES

[1] Q. Y. Li, J. B. Wang, and H. W. Wang, ''Study on impact resistance of industrial safety helmet,'' J. Saf. Sci. Technol., vol. 17, no. 3, pp. 182–186, Mar. 2021, doi: 10.11731/j.issn.1673-193x.2021.03.028.
[2] Y. X. Wang, Z. Wang, and B. Wu, ''Research review of safety helmet wearing detection algorithm in intelligent construction site,'' J. Wuhan Univ. Technol., vol. 43, no. 10, pp. 56–62, Oct. 2021, doi: 10.3963/ j.issn.1671-4431.2021.10.00.
[3] L. Jun, W. C. Dang, and P. Lihu, ''Safety helmet detection based on YOLO,'' Comput. Syst. Appl., vol. 28, no. 9, pp. 174–179, Sep. 2019, doi: 10.15888/j.cnki.csa.007065.
[4] W. Bing, L. Wenjing, and T. Huan, ''Improved YOLOv3 algorithm and its application in helmet detection,'' Comput. Eng. Appl., vol. 26, no. 9, pp. 33–40, Feb. 2020, doi: 10.3778/j.issn.1002-8331.1912-0267.
[5] F. Ming, S. Tengteng, and S. Zhen, ''Fast helmet-wearing-condition detection based on improved YOLOv2,'' Opt. Precis. Eng., vol. 27, no. 5, pp. 1196–1205, Mar. 2019, doi: 10.3788/OPE.20192705.1196.
[6] H. C. Zhao, X. X. Tian, and Z. S. Yang, ''YOLO-S: A new lightweight helmet wearing detection model,'' J. East China Normal Univ. Natural Sci., vol. 47, no. 5, pp. 134–145, Sep. 2021, doi: 10.3969/j.issn.1000- 5641.2021.05.012.
[7] T. Ding, X. Y. Chen, Q. Zhou, and H. L. Xiao, ''Real-time detection of helmet wearing based on improved YOLOX,'' Electron. Meas. Technol., vol. 45, no. 17, pp. 72–78, Sep. 2022, doi: 10.19651/j.cnki.emt.2209425.
[8] X. Ma, K. Ji, B. Xiong, L. Zhang, S. Feng, and G. Kuang, ''LightYOLOv4: An edge-device oriented target detection method for remote sensing images,'' IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens., vol. 14, pp. 10808–10820, 2021, doi: 10.1109/JSTARS.2021.3120009.
[9] Z. Z. Sun, X. G. Len, and L. Yu, ''BiFA-YOLO: A novel YOLObased method for arbitrary-oriented ship detection in high-resolution SAR images,'' Remote Sens., vol. 13, no. 21, pp. 4209–4237, Oct. 2021, doi: 10.3390/rs13214209.
[10] R. Girshick, J. Donahue, T. Darrell, and J. Malik, ''Rich feature hierarchies for accurate object detection and semantic segmentation,'' in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2014, pp. 580–587.
[11] R. Girshick, ''Fast R-CNN,'' in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Dec. 2015, pp. 1440–1448.
[12] S. Ren, K. He, R. Girshick, and J. Sun, ''Faster R-CNN: Towards realtime object detection with region proposal networks,'' IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 6, pp. 1137–1149, Jun. 2017.
[13] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, ''The PASCAL visual object classes (VOC) challenge,'' Int. J. Comput. Vis., vol. 88, no. 2, pp. 303–338, Jun. 2010.
[14] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, ''You only look once: Unified, real-time object detection,'' in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 779–788.
[15] W. Liu, D. Anguelov, and D. Erhan, ''SSD: Single shot multibox detector,'' in Proc. Eur. Conf. Comput. Vis. (ECCV), Oct. 2016, pp. 21–37.
[16] A. Howard, M. Sandler, B. Chen, W. Wang, L.-C. Chen, M. Tan, G. Chu, V. Vasudevan, Y. Zhu, R. Pang, H. Adam, and Q. Le, ''Searching for MobileNetV3,'' in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2019, pp. 1314–1324.
[17] J. Hu, L. Shen, and G. Sun, ''Squeeze-and-excitation networks,'' in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 7132–7141.
[18] Y. H. Shao, D. Zhang, and H. Y. Chu, ''A review of Yolo object detection based on deep learning,'' J. Electron. Inf. Technol., vol. 44, no. 10, pp. 3697–3708, Oct. 2022, doi: 10.11999/JEIT210790.
[19] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, ''Generalized intersection over union: A metric and a loss for bounding box regression,'' in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 658–666.
[20] A. Neubeck and L. Van Gool, ''Efficient non-maximum suppression,'' in Proc. 18th Int. Conf. Pattern Recognit. (ICPR), Aug. 2006, pp. 850–855.
[21] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, ''Path aggregation network for instance segmentation,'' in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 8759–8768.
[22] S. Woo, J. Park, and J. Y. Lee, ''CBAM: Convolutional block attention module,'' in Proc. Eur. Conf. Comput. Vis. (ECCV), Sep. 2018, pp. 3–19.

Dataset Link:

Used: need to create the dataset from https://roboflow.com/convert/labelbox-json-to-yolov5-pytorch-txt



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)