IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

SMART PPE DETECTION ENHANCING SAFETY IN INDUSTRIAL ENVIRONMENTS

[1]V.Chaitanya, [2]V.Bharathi, [3]N.Srinivas

[1]Faculty, [2]Faculty, [3]Faculty

[1]Computer Science Department,

[1]Geethanjali Institute of Science and Technology, Nellore.

Abstract: PPE stands for Personal Protective Equipment, which refers to protective gear or equipment designed to protect individuals from potential hazards or injuries in the workplace. PPE includes items such as helmets, gloves, safety glasses, vest. The use of PPE is essential in many industries, such as construction, manufacturing, and healthcare, to ensure the safety and well-being of workers. PPE must meet specific safety standards and be appropriate for the type of work being performed to provide effective protection against potential hazards. Deep learning techniques have enabled the automation of PPE detection, making the process more accurate and efficient. This paper proposes an extension to an existing PPE detection system using YOLOv5 that detects whether a person is wearing a helmet or not. The proposed system employs YOLOv8 and achieves an accuracy of 94.5% for detecting helmets, an improvement over the existing system's 92.2% accuracy. Additionally, the proposed system provides several new features, including the detection of vests, gloves, goggles, masks, and shoes. These new features enhance the system's safety capabilities by identifying workers who may be at risk due to a lack of appropriate PPE. The proposed system's performance demonstrates the potential of deep learning techniques in improving workplace safety, reducing workplace accidents, and promoting safer working environments.

Index Terms -PPE, safety and protection, computer vision, yolo model, deep learning.

I. INTRODUCTION

Industrial environments are typically workplaces where heavy machinery, equipment, and tools are used to manufacture products or provide services. These environments can be found in various industries such as manufacturing, construction, mining, and transportation. The industrial environment is often characterized by the presence of loud noise, dust, heat, hazardous chemicals, and heavy equipment. These factors can pose a risk to the health and safety of workers if not managed properly.

There are several hazards that are commonly found in industrial environments. The following are some of the most significant risks:

Falls - Falls are a leading cause of injury in industrial environments, often caused by working at heights or on uneven surfaces. Falls can result in broken bones, head injuries, and even fatalities.

Electrocution - Electrical hazards are present in most industrial environments, and electrocution can occur when workers come into contact with live wires or faulty equipment. Electrocution can cause burns, cardiac arrest, and death.

Burns - Burns can occur from exposure to hot surfaces, chemicals, or flames. Burns can result in permanent scarring and disability.

Exposure to harmful substances - Workers in industrial environments can be exposed to a wide range of hazardous substances, including toxic chemicals, gases, and fumes. Exposure to these substances can cause respiratory problems, cancers, and other illnesses.

Equipment-related accidents - Industrial environments are often characterized by the use of heavy machinery and equipment. Equipment-related accidents can result in crushing injuries, amputations, and fatalities.

Industrial safety is necessary in the current scenario because it ensures the health, safety, and welfare of workers and visitors in industrial workplaces, which can help prevent accidents, injuries, and illnesses. With the advancement of technology and the increasing complexity of industrial processes, the potential for accidents and hazards has also increased. Industrial workplaces are often characterized by the presence of heavy machinery, hazardous materials, and other potential hazards, which can cause serious harm to workers and visitors if not managed properly.

Moreover, ensuring industrial safety is not only important for the well-being of workers and visitors, but it also benefits the organization itself. Accidents and injuries can result in lost productivity, increased insurance costs, and legal liabilities. On the other hand, a safe work environment can increase productivity, reduce absenteeism, and improve the morale and motivation of workers.

Additionally, in the current scenario, the COVID-19 pandemic has added a new dimension to industrial safety, with employers needing to implement measures to prevent the spread of infectious diseases in the workplace. This includes measures such as social distancing, use of personal protective equipment, regular cleaning and disinfecting, and other safety measures.

In the present scenario, there are several safety measures that can be taken in order to ensure industrial safety. Here are some of them.Conduct a risk assessment: Before starting any industrial process, it is important to assess the potential risks involved in the process. This will help in identifying the hazards and taking measures to prevent accidents.Provide adequate training: Employees should be provided with adequate training on safety measures and precautions to be taken in the workplace. This will help them to handle the equipment and machinery safely.Use Personal Protective Equipment (PPE): Employees should wear appropriate PPE such as helmets, gloves, goggles, etc. depending on the type of work they are doing.

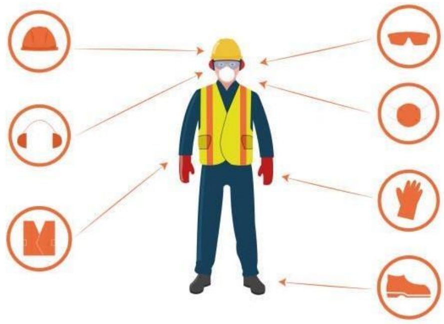

Fig 1 Personal Protective Equipment

Ensure equipment maintenance: All equipment should be maintained and inspected regularly to ensure that they are functioning properly. Faulty equipment can lead to accidents and injuries.

Implement safety procedures: Safety procedures should be implemented and communicated to all employees. These procedures should be followed strictly to avoid accidents. Provide emergency preparedness training: Employees should be trained to respond to emergencies such as fire, spills, and other incidents. This will help them to act quickly and efficiently in such situations. Encourage reporting of hazards: Employees should be encouraged to report any hazards or unsafe conditions that they observe in the workplace. This will help in identifying potential hazards and taking measures to prevent accidents.

YOLOv8 is a powerful computer vision algorithm that can be used for object detection, including Personal Protective Equipment (PPE) in industrial settings. In industrial safety, YOLOv8 can be used to improve worker safety by detecting instances of non-compliance with PPE requirements.

The first step in using YOLOv8 for PPE detection is to create a dataset of images or videos of workers in industrial settings. The dataset should include examples of workers wearing different types of PPE, as well as examples of workers not wearing PPE. Once the dataset is created, the algorithm can be trained to recognize and differentiate between different types of PPE.

Once the algorithm is trained, it can be used in real-time to detect PPE non-compliance. Cameras placed in industrial settings can capture images or videos of workers, which are then analyzed by the algorithm to determine whether workers are wearing the appropriate PPE. When non-compliance is detected, the algorithm can alert safety personnel or supervisors, who can take corrective action.

II. LITERATURE REVIEW

"Deep learning-based safety helmet detection in engineering management based on convolutional neural networks": The paper proposes a deep learning-based approach for detecting safety helmets worn by workers in construction sites. A convolutional neural network model is trained on a large dataset of annotated images to detect the presence or absence of safety helmets. The proposed method achieves high accuracy and performance in real-time helmet detection, with potential applications in engineering management and workplace safety.

"Detecting safety helmet wearing on construction sites with bounding-box regression and deep transfer learning": This paper introduces a method for detecting safety helmet wearing in construction sites using bounding-box regression and deep transfer learning. The approach involves locating helmets on workers' heads through video monitoring and training a model using limited data. The method achieves high accuracy in detecting safety helmet wearing and offers potential benefits in improving safety and compliance on construction sites.

"An analysis on safety risk judgment patterns towards computer vision-based construction safety management": This study analyzes the perceptions and judgments of safety risks among construction workers and their attitudes towards computer vision-based safety management systems. The research identifies workers' preferences for physical factors over behavioral factors in safety risk judgments and highlights the importance of considering these patterns in developing computer vision-based safety systems.

"Safety helmet wearing detection method of improved YOLOv3": The paper proposes an improved method for detecting safety helmet wearing using the YOLOv3 object detection model. The approach involves retraining the pre-trained model on a custom dataset of helmet images and addressing class imbalance through oversampling and weighted loss. The method shows promise for safety helmet detection in practical scenarios.

"YOLOv4: Optimal speed and accuracy of object detection": This paper introduces YOLOv4, a state-of-the-art object detection model that achieves high accuracy and real-time processing speeds. The authors propose modifications to the YOLOv3 architecture and training techniques, resulting in improved performance on benchmark datasets. The paper provides detailed analysis and comparisons with other models, offering insights into the capabilities of YOLOv4 for object detection tasks.

III. PROPOSED METHODOLOGY

YOLOv8 (You Only Look Once version 8) is a deep learning model for object detection that uses a single neural network to perform both object localization and classification. YOLOv8 uses a variant of the YOLO architecture called "Scaled-YOLOv8," which is designed to be more efficient and accurate than previous versions.

In YOLOv8, the input image is divided into a grid of cells, and each cell is responsible for detecting objects within its boundaries. For each object detected, the model outputs the bounding box coordinates (i.e., the top-left and bottom-right corners of the box) and the class probabilities (i.e., the probability that the object belongs to each class in the dataset).

Fig 1: Schematic diagram of image division

The detection process in YOLOv8 is performed in multiple stages. First, the input image is processed by a series of convolutional layers that extract features from the image. These features are then used to generate a set of candidates bounding boxes at various locations and scales within the image. Next, the model applies a non-maximum suppression (NMS) algorithm to remove redundant boxes and keep only the most confident detections.Finally, the model performs class prediction by assigning a probability to each detected box for each class in the dataset. The class with the highest probability is assigned to the box as the predicted class label.Overall, YOLOv8's approach to object detection is efficient and accurate, making it suitable for a wide range of applications, including PPE detection in industrial environments.

Comparison of Models :

The following table gives the algorithm used, accuracy, classes and FPS. Starting with algorithm used by existing system is YOLOV5, accuracy for detecting the helmet is 92.2%,FPS is around 65FPS. While the proposed system includes the algorithm YOLOV8, accuracy for detecting the helmet is 93.3% and there extra classes we are detecting here like helmet, vest, gloves, googles, shoes and mask and the FPS for detecting the frames in video is 160 frames per second.

Model	Algorithm	Accuracy	Classes	FPS
Existing System	YOLOV5	92.2%	HELMET	65-80
Proposed System	YOLOV8	94.5%	HELMET, VEST, GLOVE, GOOGLES, SHOES, MASK	160

Table 1: Comparison of Existing and Proposed system

The modules involved in our proposed system are Data collection, Data pre-processing, Training the model, Validating the model and testing the model.

DATA COLLECTION:

Data collection is a crucial step in PPE detection. There are two main ways to collect data for PPE detection:

Manual Data Collection: This involves manually collecting images or videos of people wearing PPE. This can be time-consuming and may not result in a large enough dataset for effective training.

Data Collection using Roboflow: Roboflow is an online platform that automates the process of collecting and labelling data for machine learning tasks. It can be used to collect and label images or videos of people wearing PPE. Roboflow allows you to upload your own images or videos, or use publicly available datasets. Once the data is uploaded, Roboflow can automatically label objects in the images or videos, making it faster and more efficient to collect and label large amounts of data.

DATA PRE-PROCESSING:

Data pre-processing is an important step in PPE detection to ensure that the data is properly prepared for training. The following are some common data pre-processing steps in PPE detection:

Image resizing and cropping: The images may need to be resized and cropped to a specific size to ensure they are compatible with the training algorithm. The images may also need to be cropped to focus on the area of the image where the PPE is located.

Image augmentation: Image augmentation techniques can be used to increase the size of the dataset by generating new images from the existing images. Augmentation techniques can include flipping, rotating, zooming, and changing the brightness and contrast of the image.

Object labelling: The images need to be labelled to identify the location and type of PPE. This can be done manually or using a tool like Roboflow to automatically label the objects in the images.

Data normalization: The pixel values of the images may need to be normalized to ensure that they fall within a certain range. This can help improve the performance of the training algorithm.

Data splitting: The dataset can be split into training, validation, and testing sets. The training set is used to train the model, the validation set is used to tune the model's hyperparameters, and the testing set is used to evaluate the model's performance on new data.

Training a PPE detection model using YOLOv8 involves the following steps :

Install YOLOv8: The first step is to install the YOLOv8 framework on your computer. You can install it using pip using the following command.

```
pip install -q ultralytics
```

Prepare the dataset: Before training the model, you need to prepare the dataset by resizing and augmenting the images, labeling the objects in the images, and splitting the dataset into training, validation, and testing sets in the previous step, Data pre-processing. We need to run the following command after successfully Pre-processing.

Create a configuration file: You need to create a configuration file that specifies the model architecture, training hyperparameters, and the paths to the training, validation, and testing sets.

Train the model: Once the configuration file is set up, you can start training the model using the YOLOv8 framework. The training process involves feeding batches of images through the model and adjusting the model's parameters to minimize the loss function.

Validating the Model:

Validation is an important step in the PPE detection pipeline using YOLOv8. It helps you to ensure that the model is accurate and reliable in detecting PPE in new, unseen images. Here are the steps to validate the model in PPE detection using YOLOv8:

- Prepare the validation dataset: Prepare a separate validation dataset that consists of a subset of images from the original dataset. These images should not have been used for training the model.
- Run validation script: YOLOv8 comes with a validation script that can be used to evaluate the model's performance on the validation dataset. Run the following command to evaluate the model on the validation set:

  python detect.py --weights <path to trained weights> --img <image size> --conf

  <confidence threshold> --source <path to validation dataset>.

  This will output the precision, recall, and mAP (mean average precision) of the model on the validation dataset.
- Analyze the results: Analyze the output of the validation script to determine the model's performance on the validation set. Look at the precision, recall, and mAP scores to see how well the model is detecting PPE.
- Fine-tune the model: If the model's performance on the validation set is not satisfactory, you can fine-tune the model by adjusting the hyperparameters or by training it on a larger dataset.
- Repeat the validation: After fine-tuning the model, repeat the validation step to evaluate the model's performance again. Repeat this process until the model performs well on the validation set.

Testing the model:

Testing the PPE detection model using a Graphical User Interface (GUI) can make it easier to interact with the model and visualize its results. Here are the steps to test the model using a GUI:

- Choose a GUI tool: There are several GUI tools available for testing YOLOv8 models, such as LabelImg,
- Load the trained model: Load the trained YOLOv8 model into the GUI tool. Make sure to specify the correct path to the model's weights and configuration files.
- Select an image or video: Select an image or video that you want to test the model on. You can either select an image from your computer or capture a video from your webcam.
- Run the model: Run the model on the selected image or video using the GUI tool. The model will detect PPE in the image or video and highlight them with bounding boxes
- Analyze the results: Analyze the output of the model to see how well it is detecting PPE. Look at the number of false positives and false negatives to determine the accuracy of the model.
- Fine-tune the model: If the model's performance is not satisfactory, you can fine-tune the model by adjusting the hyperparameters or by training it on a larger dataset.
- Repeat the testing: After fine-tuning the model, repeat the testing step to evaluate the model's performance again. Repeat this process until the model performs well on the test data.

Streamlit is an open-source Python library used for building and deploying data-driven web applications. It simplifies the process of creating interactive and user-friendly interfaces for data analysis, machine learning, and visualization. Streamlit allows developers and data scientists to focus on the core functionality of their applications without worrying about the underlying web development aspects. Streamlit is widely used by data scientists, machine learning engineers, and developers to create interactive dashboards, data exploration tools, and machine learning prototypes. Its simplicity, rapid development workflow, and intuitive interface make it a popular choice for building data-driven applications.

The actual implementation of the system goes as follows.First the dataset if collected from roboflow platform which is preprocessed and read to train dataset.It is the loaded into google colab platform and by using ultralytics library we import yolo and yolo pretrained model.We train on the custom dataset with different category images.Then a model is saved and linked with the streamlit interface which looks like in the Fig 2.

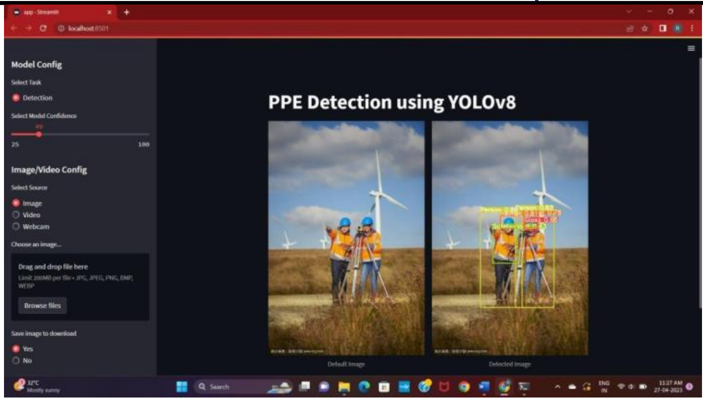

Fig 2. Streamlit webpage

IV. CONCLUSION

In conclusion, the use of YOLOv8-based smart PPE detection technology has the potential to significantly enhance safety in industrial environments. By utilizing deep learning algorithms, YOLOv8-based systems can detect the use of personal protective equipment (PPE) by workers in real-time, ensuring that all workers are properly equipped to handle potential hazards.

The implementation of smart PPE detection technology using YOLOv8 can improve workplace safety by providing immediate alerts and warnings to workers and supervisors when PPE is not being used correctly. This can help reduce the risk of accidents and injuries, while also improving overall safety and productivity in industrial settings.

Furthermore, the use of YOLOv8-based systems can also help companies comply with safety regulations and reduce liability risks. The data collected by these systems can be used to identify patterns and trends in safety incidents, leading to targeted improvements in safety protocols and procedures.

REFERENCES

1) R. A. Haslam, S. A. Hide, A. G. F. Gibb, D. E. Gyi, T. Pavitt, S. Atkinson, and A. R. Duff, ‘‘Contributing factors in construction accidents,’’ Appl. Ergonom., vol. 36, no. 4, pp. 401– 415, 2005.
2) S. Rowlinson, Construction Safety Management Systems. Evanston, IL, USA: Routledge, 2004.
3) N. Cavazza and A. Serpe, ‘‘Effects of safety climate on safety norm violations: Exploring the mediating role of attitudinal ambivalence toward personal protective equipment,’’ J. Saf. Res., vol. 40, no. 4, pp. 277–283, Aug. 2009.
4) J. Wu, N. Liu, C. Geyer, and J. M. Rehg, ‘‘C4 : A real-time object detection framework,’’ IEEE Trans. Image Process., vol. 22, no. 10, pp. 4096–4107, Jun. 2013.
5) S. Y. Nikouei, Y. Chen, and T. R. Faughnan, ‘‘Smart surveillance as an edge service for real-time human detection and tracking,’’ in Proc. IEEE/ACM Symp. Edge Comput. (SEC), Oct. 2018, pp. 336–337
6) B. Balakreshnan, G. Richards, G. Nanda, H. Mao, R. Athinarayanan, and J. Zaccaria, ‘‘PPE compliance detection using artificial intelligence in learning factories,’’ Proc. Manuf., vol. 45, pp. 277–282, Jan. 2020
7) W. Z. Khan, E. Ahmed, S. Hakak, I. Yaqoob, and A. Ahmed, ‘‘Edge computing: A survey,’’ Future Gener. Comput. Syst., vol. 97, pp. 219–235, Aug. 2019.
8) A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, ‘‘YOLOv4: Optimal speed and accuracy of object detection,’’ Tech. Rep., 2020.
9) M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, ‘‘MobileNetV2: Inverted residuals and linear bottlenecks,’’ in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 4510–4520
11) M. Tan, R. Pang, and Q. V. Le, ‘‘EfficientDet: Scalable and efficient object detection,’’ in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 10781– 10790.
12) G. Gallo, F. Di Rienzo, P. Ducange, V. Ferrari, A. Tognetti, and C. Vallati, ‘‘A smart system for personal protective equipment detection in industrial environments based on deep learning,’’ in Proc. IEEE Int. Conf. Smart Comput. (SMARTCOMP), Aug. 2021, pp. 222–227
13) X. Luo, H. Li, H. Wang, Z. Wu, F. Dai, and D. Cao, ‘‘Vision-based detection and visualization of dynamic workspaces,’’ Autom. Construct., vol. 104, pp. 1–13, Aug. 2019
14) B. E. Mneymneh, M. Abbas, and H. Khoury, ‘‘Vision-based framework for intelligent monitoring of hardhat wearing on construction sites,’’ J. Comput. Civil Eng., vol. 33, no. 2, Mar. 2019, Art. no. 0
15)] J. Yang, S. Li, Z. Wang, and G. Yang, ‘‘Real-time tiny part defect detection system in manufacturing using deep learning,’’ IEEE Access, vol. 7, pp. 89278–89291, 2019.
16) Q. Fang, H. Li, X. Luo, L. Ding, H. Luo, and C. Li, ‘‘Computer vision aided inspection on falling prevention measures for steeplejacks in an aerial environment,’’ Autom. Construct., vol. 93, pp. 148–164, Sep. 2018.
17) W. Zhiqiang and L. Jun, ‘‘A review of object detection based on convolutional neural network,’’ in Proc. 36th Chin. Control Conf. (CCC), Jul. 2017, pp. 11104–11109.
18) 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
19) R. Girshick, ‘‘Fast R-CNN,’’ in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Dec. 2015, pp. 1440–1448
20) R. Girshick, J. Donahue, T. Darrell, and J. Malik, ‘‘Rich feature hierarchies for accurate object detection and semantic segmentation,’’ in Proc. IEEE
21) 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.

22) W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Fu, and A. C. Berg, ''SSD:Single shot MultiBox detector,'' in Computer VisionECCV, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham, Switzerland: Springer, 2016, pp. 21–37.
23) 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.
24) J. Redmon and A. Farhadi, ''YOLOv3: An incremental improvement,'' in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Apr. 2018, pp. 1–6.
25) A. Bochkovskiy, C.-Y. Wang, and H.-Y. Liao, ''YOLOv4: Optimal speed and accuracy of object detection,'' in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Apr. 2020, pp. 1– 17.
26) Y. Zhou, H. Xue, X. Jiang, H. Sun, and X. Yanyan, ''Low-resolution safety helmet image recognition combining local binary pattern with statistical features,'' Comput. Syst. Appl., vol. 24, no. 7, pp. 211–215, 2015.
27) B. E. Mneymneh, M. Abbas, and H. Khoury, ''Automated hardhat detection for construction safety applications,'' Proc. Eng., vol. 196, pp. 895–902, Jan. 2017.
28) Y. Gu, S. Xu, Y. Wang, and L. Shi, ''An advanced deep learning approach for safety helmet wearing detection,'' in Proc. Int. Conf. Internet Things (iThings), IEEE Green Comput. Commun. (GreenCom), IEEE Cyber, Phys. Social Comput. (CPSCom), IEEE Smart Data (SmartData), Jul. 2019, pp. 669–674.
29) F. Wu, G. Jin, M. Gao, Z. He, and Y. Yang, ''Helmet detection based on improved YOLOv3 deep model,'' in Proc. IEEE 16th Int. Conf. Netw., Sens. Control (ICNSC), May 2019, pp. 363–368.
30) G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, ''Densely connected convolutional networks,'' in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 2261–2269.