Micro Laboratory Safety Hazard Detection Based on YOLOv4: A Lightweight Image Analysis Approach

Yuan Lin*
School of Chemistry and Chemical Engineering, Hainan University, Haikou, Hainan, China

***Abstract*—In hazardous chemical laboratories, identifying and managing safety hazards is critical for effective safety management. This study, grounded in safety engineering principles, focuses on laboratory environments to develop an efficient hazard detection model using deep learning and object detection techniques. The lightweight YOLOv4-Tiny algorithm, with fewer parameters, was selected and optimized for detecting unsafe factors in laboratories. The CIOU loss function was employed to enhance the stability of candidate box regression, while three attention mechanism modules were embedded into the backbone feature extraction network and the feature pyramid's upsampling layer, forming an improved YOLOv4-Tiny object detection algorithm. To support the detection tasks, a specialized dataset for laboratory hazards was created. The improved YOLOv4-Tiny model was then used to construct two detection models: one for identifying the status of chemical bottles and another for detecting general laboratory safety hazards. The chemical bottle status detection model achieved AP values of 93.06% (normal), 95.31% (disorderly stacking), and 90.72% (label detachment), with an mAP of 93.03% and an FPS of 272, demonstrating both high accuracy and speed. The laboratory hazard detection model achieved AP values of 97.40%, 90.14%, 96.80%, and 68.95% for normal experimenters, individuals not wearing protective equipment, individuals smoking, and open flames, respectively, with a mAP of 88.32% and an FPS of 116. These results confirm the effectiveness of the proposed models in accurately and efficiently identifying laboratory safety hazards.**

***Keywords*—Hazardous chemical safety; unsafe factors; deep learning; target detection; YOLO-v4-tiny; laboratory safety**

I. Introduction

According to statistics, laboratory accidents have accounted for 20% of safety incidents over the past century, second only to fire accidents. The chemicals and equipment used in laboratories are essential components of scientific research, supporting the development of related fields. However, the toxic, flammable, explosive, and corrosive properties of chemicals make laboratories prone to accidents such as poisoning, fires, explosions, and injuries during daily operations. Incomplete statistics show that globally, from 2015 to 2024, there were 5,513 laboratory safety accidents, resulting in 5,592 injuries and 2,560 deaths. This indicates that the safety situation in laboratories is quite severe, with frequent accidents not only hindering the smooth progress of research but also threatening the safety of laboratory personnel. Therefore, researching emerging technologies to improve laboratory safety management is of great practical significance.

The direct cause of accidents resulting in casualties is the presence of unsafe factors, specifically unsafe behaviors of personnel and unsafe conditions of equipment. Therefore, the key to preventing accidents lies in eliminating these unsafe factors. Traditional safety management relies on manual monitoring, which is not only inefficient but also passive.

As machine learning, neural networks, and deep learning technologies mature, various industries are gradually moving towards informatization and intelligent development. In recent years, laboratory safety management technology has seen significant development opportunities. Intelligent safety management technologies have continuously emerged and been successfully applied in practical work, such as safety helmet detection and fall hazard warnings. The successful application of artificial intelligence in these areas has demonstrated its effectiveness in improving safety levels.

Therefore, researching deep learning-based methods for detecting unsafe factors is crucial for enhancing the efficiency of laboratory safety management, speeding up accident response times, reducing the likelihood of accidents, and strengthening accident rescue capabilities.

1) Unclear detection targets: Laboratory accidents are varied, including fires, explosions, injuries, poisoning, and electric shocks. Accidents often result from the combined effect of multiple factors, characterized by complexity, randomness, and suddenness. However, most existing technical solutions focus only on individual unsafe factors, lacking a systematic analysis and detection of overall unsafe factors in laboratories.

2) Insufficient unsafe factor image datasets: The complexity and diversity of laboratory accidents lead to varying forms of unsafe factors, making the design and collection of image data challenging. The lack of unsafe factor image datasets is a pressing problem that needs to be addressed.

3) Detection models need to meet requirements for real-time, accuracy, and stability: The complex and changing laboratory environment imposes higher demands on the performance of detection algorithms. Due to the sudden nature of accidents, detection models must have real-time capabilities and high accuracy to promptly identify and handle unsafe factors, preventing accidents.

To address the above issues, this paper selects the YOLO-v4-tiny algorithm, which has smaller model parameters, for conducting research on laboratory unsafe factor detection. Subsequently, a dataset of laboratory unsafe factors was

established to verify that this method can detect unsafe factors while meeting the requirements for detection accuracy and speed. Section II summarizes related work on object detection, Section III proposes an improved object detection model, Section IV verifies the effectiveness of this method through experiments, and Section V concludes the effectiveness of this method.

II. Literature Review

This paper will collect existing work on automated target detection algorithm to highlight the shortcomings of existing research.

A. Traditional Target Detection Algorithm

Traditional object detection algorithms typically operate by analyzing the motion characteristics of objects, designing feature operators, and extracting these features from the frames to be analyzed. These algorithms often have complex structures, leading to low detection speeds and limited recognition accuracy. Viola and colleagues [1] [2] made a significant breakthrough by designing a model that achieved real-time face detection for the first time. Their model employed a sliding window detection method, extracting features of various sizes from different positions within the detection frames, and then using classifiers to categorize the objects. Due to the high computational demands of this approach, which exceeded the capabilities of computers at the time, the model incorporated techniques like "integral images" and "detection cascades" to optimize performance and enhance detection speed.

To further improve detection speed and address the trade-off between feature invariance and non-linearity in object detection tasks, Dalal and colleagues [3] introduced the Histogram of Oriented Gradients (HOG) descriptor. HOG was primarily designed for pedestrian detection, allowing the input image to be rescaled multiple times while keeping the candidate boxes at a fixed size, thereby achieving effective detection.

The Deformable Part-based Model (DPM), proposed by Felzenszwalb and colleagues [4] [5], represents the apex of traditional object detection methods. The core concept of DPM involves segmenting the object into parts, such as detecting components like wheels and windshields when identifying a car. Building on DPM, Girshick and colleagues [6] integrated a cascade structure into the model, optimizing it to significantly increase detection speed—up to ten times faster—without sacrificing accuracy. This enhanced DPM model marked the peak of traditional object detection techniques in terms of both accuracy and speed.

However, with the continuous advancements in computer parallel processing capabilities, deep learning-based object detection models have gradually surpassed traditional methods, offering superior detection accuracy and speed.

B. Object Detection Algorithm Based on Convolutional Neural Network

The predecessor of Convolutional Neural Networks (CNNs) was the structure proposed by Fukushima, which included pooling and convolutional layers [7]. Building on this, Lecun introduced the backpropagation algorithm, forming the basic architecture of CNNs [8]. However, due to the limited computational power at the time, CNNs did not gain widespread application.

The AlexNet model, proposed by Hinton's team, won the image classification competition, demonstrating the powerful image processing capabilities of CNNs [9]. The AlexNet network consists of three fully connected layers and five convolutional layers, using ReLU as the activation function and Dropout to prevent overfitting, achieving a test error rate of only 15.3%. This success sparked widespread interest in applying CNNs to image processing tasks. Subsequently, Simonyan and others proposed VGG-Net, which deepened the network layers (16-19 layers) and used smaller convolutional kernels, reducing the error rate to 7.3% [10]. GoogLeNet further optimized the network structure by introducing the Inception module, enhancing detection performance without excessively increasing model parameters. In 2015, Kaiming He proposed ResNet, which solved the vanishing and exploding gradient problems in deep networks through a residual structure, allowing the network layers to exceed 1,000[11].

CNN-based object detection algorithms are mainly divided into Two-stage and One-stage methods. Girshick proposed RCNN, the first deep learning-based object detection algorithm, marking a significant advancement in object detection [12]. Subsequently, Fast RCNN and Faster RCNN further optimized detection speed and model performance [13]. Unlike Two-stage methods, One-stage algorithms like YOLO can directly perform feature extraction, classification, and localization through CNNs, significantly improving detection speed [14]. The YOLO series algorithms have continued to evolve, with YOLO-v2 improving the network structure and enhancing the model's mAP [15], and YOLO-v3 further improving detection accuracy [16].

From the above discussion, it is evident that single-stage object detection algorithms have become mainstream. This paper constructs an unsafe factor detection model based on the YOLO series algorithms.

C. Research Gaps

While the use of artificial intelligence in managing the safety of hazardous chemical storage and usage has become an industry trend, research specifically focused on chemical laboratory management remains underdeveloped. The primary challenges include:

1) Unclear detection targets: Current studies mainly address the management and safety of hazardous chemicals during transportation, lacking a systematic framework for identifying unsafe factors within chemical laboratories. Accidents in these labs—such as fires, explosions, and poisonings—are complex and varied, requiring a comprehensive analysis to pinpoint key unsafe factors.

2) Insufficient image datasets: The unique operations and technologies in chemical laboratories lead to diverse unsafe scenarios, making data collection challenging. Existing datasets are inadequate for fully training and optimizing target

detection models in this context.

3) Model performance requirements: Although convolutional neural network-based detection technologies have shown promise, their application in chemical labs demands higher real-time performance, accuracy, and stability. The unpredictability of accidents necessitates that detection models effectively identify and address unsafe factors in real-time to prevent incidents.

In summary, this paper focuses on chemical laboratories as a key area for hazardous chemical management. It aims to analyze accident types and causes using safety system analysis methods, identify specific hazard sources and risk levels, and customize and optimize a target detection model for the accurate identification of key unsafe factors.

III. Improved Object Detection Model Based on Attention Mechanism

If unsafe factors arise in a chemical laboratory, accidents can easily occur, leading to casualties. Therefore, it is crucial to control these factors before an accident happens. To enable the rapid and accurate identification of unsafe factors in chemical laboratories, this paper integrates an attention mechanism into the lightweight YOLO-v4-tiny model, further enhancing detection accuracy and speed, thereby laying the foundation for the identification and detection of such factors.

A. YOLO-v4-Tiny Algorithm

The YOLO-V4- tiny is a simplified version of the YOLO-v4 algorithm, although the detection accuracy is slightly inferior, but because of the simplification of the structure, its model parameters are reduced from 60 million to 6 million, which is more suitable for engineering applications.

The backbone feature extraction network of YOLO-v4-tiny is CSPDarkNet53-Tiny. In addition to the network structure, the improvements of CSPDarkNet53-Tiny mainly include: changing the activation function of the convolutional network from LeakyReLU to Mish; the residual network structure is optimized to CSPnet.

The formula for Mish activation function is:

$$\text{Mish} = x \times tanh\left(ln(1 + e^{x})\right) \quad (1)$$

Mish indicates the output of the activation function; x represents input.

The YOLO-v4 network uses the LeakyReLU activation function. The Mish activation function versus the LeakyReLU function is shown in Fig. 1. As can be seen from Fig. 1, compared to LeakyReLU function, Mish function is smoother, allowing the network to mine deeper feature information. And unlike ReLU, which takes 0 directly in the negative region, Mish function is smoother at 0 and has better gradient flow towards negative values, thus making the model more accurate.

The YOLO-v4-tiny model adjusts the original residual structure and uses the CSPnet residual structure. The CSPnet residual structure is shown in Fig. 2.

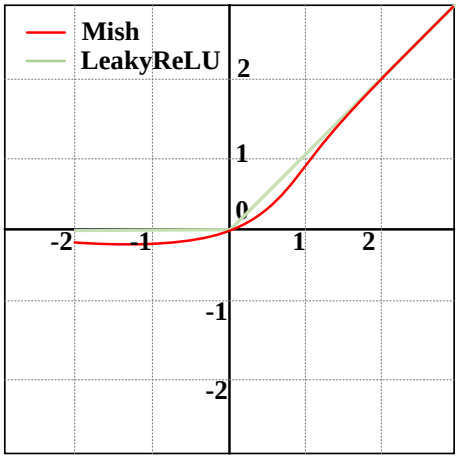

Fig. 1. Mish loss function.

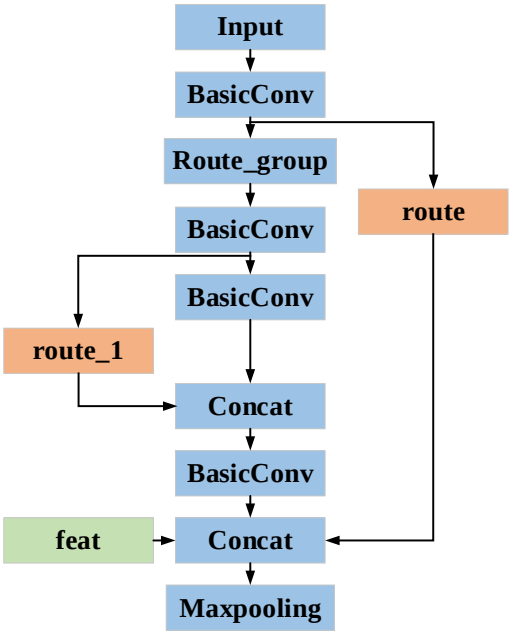

Fig. 2. CSPnet residual structure.

In the CSPnet structure, after the input of the feature layer (h,w,c), a convolution operation is performed first, and then the feature layer in the input network is divided into two parts (route) in the channel. The trunk part is further divided into two parts in the channel after a convolution operation. The trunk is merged with branch route_1 after one convolution operation, and the merged feature layer is merged with branch route and feat after one convolution operation. Finally, a maximum pooling operation is performed on the feature layer to obtain the processed feature layer (h/2, w/2, 2c).

1) CSPDarkNet53-Tiny: The backbone feature extraction network of YOLO-v4-tiny model, CSPDarkNet53-Tiny, has better feature extraction capability and faster computation speed. CSPDarkNet53-Tiny consists of three basic convolution blocks and three CSPnet modules, as shown in Table I.

2) Mosaic data augmentation: The Mosaic is to stitch together four images into a single image, with the goal of enriching the background of detection targets and enhancing the model's generalization ability. The implementation method involves reading four images at once during model training, placing the augmented images in the four corners, and combining them into a new image.

TABLE I NETWORK STRUCTURE OF CSPDARKNET53-TINY

Convolution Module	Step	Number of Channels	Input	Output
Input			416×416×3	416×416×3
Convolution Block	2	32	416×416×3	208×208×32
Convolution Block	2	64	208×208×32	104×104×64
CSPnet Residual Block			104×104×64	52×52×128
CSPnet Residual Block			52×52×128	26×26×256
out1				26×26×256
CSPnet Residual Block			26×26×256	13×13×512
Convolution Block	1	512	13×13×512	13×13×512
out2				13×13×512

B. Feature Pyramid of YOLO-v4-tiny

Feature pyramid is a component of convolutional neural network which is convenient for model to detect objects of different scales. Its typical feature has a top-down structure, which is convenient for model to extract high-level semantic features on the feature layer. The YOLO-v4tiny model simplifies the feature pyramid and fuses the two feature layers output by the backbone feature extraction network. Its structure is shown in Fig. 3.

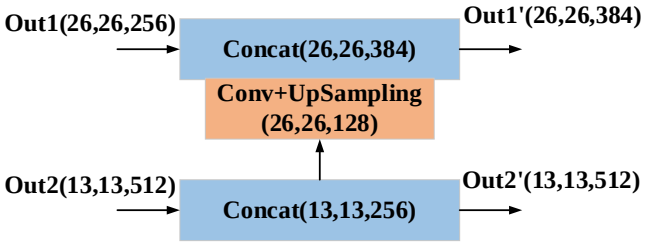

Fig. 3. YOLO-v4-tiny feature pyramid.

Feature layer out2 after input feature pyramid, a layer of convolution operation is performed to obtain feature layer out2 ^' (13, 13, 256), and feature layer out2' is used for input YOLO Head for target detection. Feature layer out2' also needs to undergo up-sampling operation to obtain feature layer with dimensions (26,26,128). Feature layer out 1(26,26,256) input feature pyramid and merge into new feature layer out 1' (26,26, 384) on channel through CONCAT operation. The feature layer out 1' is used to input YOLO Head for target detection.

C. Improved YOLO-v4-tiny Algorithm

1) CIOU: Unlike IOU, which only focuses on the overlap rate between candidate boxes and real boxes, CIOU is optimized based on IOU. It considers the overlap rate, scale, penalty term and so on between the candidate frame and the real frame, which makes the regression of the candidate frame more stable. CIOU's formula is as follows:

$$v = \frac{4}{\pi^2}\left(arctan\frac{w^{gt}}{h^{gt}} - arctan\frac{w}{h}\right)^2 \quad (2)$$

$$\alpha = \frac{v}{1-IOU+v} \quad (3)$$

$$\text{CIOU} = IOU - \frac{\rho^2(b,b^{gt})}{c^2} - \alpha v \quad (4)$$

Where, c is the maximum distance between the point on the prediction box and the point on the real box, w is the width of the image, h is the height of the image, v is the similarity, w^{gt}t represents the median value of the image width, h^{gt} represents the median value of the image height, $\rho^2(b,b^{gt})$ represents the Euclidean distance between the center points of the two boxes.

2) Loss function of YOLO-v4-tiny model: The loss function of YOLO-v4-tiny model was established based on CIOU, and the formula of the loss function of the model was obtained as follows:

$$Loss_{\text{CIOU}} = 1 - IOU + \frac{\rho^2(b,b^{gt})}{c^2} + \alpha v \quad (5)$$

3) The overall structure of YOLO-v4-tiny model with improved attention mechanism: The overall structure of YOLO-V4-tiny model includes backbone feature extraction network CSPDarknet53-Tiny, feature pyramid, attention mechanism module and feature prediction module YOLO Head. Three attention modules are embedded in the model, in which two attention mechanism modules are embedded after two output feature layers of the feature extraction network in the backbone of the YOLO-v4-tiny model, and the attention mechanism module is inserted after sampling layers on the feature pyramid. The model structure is shown in Fig. 4.

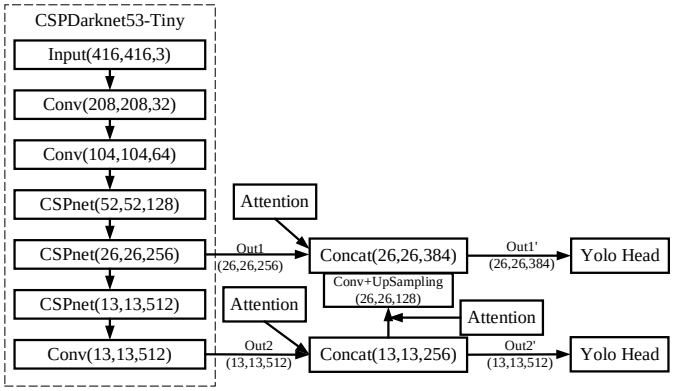

Fig. 4. An improved YOLO-v4-tiny model of attention mechanism.

IV. Experiment and Verification

In this section, the reliability and validity of the proposed method is verified through experiments.

A. Experimental Environment

In the process of establishing the laboratory dataset, personnel must first apply for access to the lab and can only proceed with experiments once they have obtained permission. The unsafe factor detection model is activated as soon as the personnel enter the laboratory, capturing one frame per second for detection. The model is designed to identify and label both normal and abnormal conditions, and it triggers an alarm if no personnel are detected. Key detection points include the use of safety gear, smoking behavior, the presence of open flames, improper storage of chemical bottles, and missing labels on bottles. Normal conditions are labeled as "Normal," while abnormal conditions are categorized based on the specific issue, such as "Fault," "Smoke," "Fire," or "Mis-drug." Typical abnormal states detected by the model are illustrated in Fig. 5. This data collection and recognition process is crucial for effective laboratory safety management.

Problem with the placement of medication bottles

Open flames appear in the laboratory

Fig. 5. An improved YOLO-v4-tiny model of attention mechanism.

Image data of unsafe factors in the laboratory were collected through on-site collection and network retrieval, and various kinds of original image data collected were shown in Table II:

TABLE II Laboratory Unsafe Factors Image Data

Detection category	quantity	Detection category	quantity
Normal	200	Fault	200
Smoke	200	Nor-drug	200
Fire	200	Mis-drug	200
Mix-drug	200	ALL	1400

The original image data of the laboratory comes from online retrieval, field capture, simulation shooting, etc. Due to different image sources and formats, it is necessary to use OpenCV computer vision library to capture the original data in JPG format, and the unified size is 416×416. After that, the image, affine, noise and other operations in the data enhancement method were used to increase the laboratory image data, and 5,600 laboratory image data were obtained. After renaming, de-reweighting, scrambling and labeling 5600 image data, the laboratory unsafe factor image dataset was constructed. The laboratory unsafe factors detection model adopts VOC data format. The data set was divided into training set, test set and verification set according to 7:2:1, and 3920 training set data, 1120 test set data and 560 verification set data were obtained. Store the image data in the JPEGImages folder and the xml file in the Annotation folder.

B. Testing Program

The experimental environment parameters of the laboratory unsafe factors identification and detection model are shown in Table III.

TABLE III Training Environment of Laboratory Unsafe Factors Identification and Detection Model

Equipment	Model (version)
Operating system	Windows10
CPU	Inter Core i7-10875H
GPU	RTX3060
CUDA	CUDA 10.0.1
cuDNN	cuDNN 7.0.5
Deep learning module	PyTorch 1.6.0
Scientific computing module	numpy 1.18.5
Computer vision module opencv	opencv-python 4.6.0

In this section, the control variable method is used for repeated experiments to determine the hyperparameters of the neural network, as shown in Table IV.

TABLE IV Laboratory Unsafe Factors Identification and Detection Model Hyperparameters

Hyperparameter type	Model hyperparameter values
Number of activations	Mish activation function
Initial learning rate	1e-2
epoch	1000
batch_size	32
Cost function	Loss

In order to accelerate the training speed of the improved YOLO-v4-tiny model, the transfer learning training method is adopted, and the training weights of coco data set are taken as pre-training weights. The detection targets of the model were not wearing safety protective equipment, smoking behavior, Normal experimental personnel, and open Fire, which were labeled as Fault, Smoke, normal, and fire respectively. One-hot coding was performed for different detection categories in the laboratory, as shown in Table V.

TABLE V ONE-HOT CODING OF THE TEST CATEGORIES IN THE LABORATORY

Detection category	Fault	Smoke	Normal	Fire
One-hot indicates	(1,0,0,0)	(0,1,0,0)	(0,0,1,0)	(0,0,0,1)

C. Analysis of Drug Status Testing Results

The medicine bottle state detection model trained 1000 EPOchs in total, and the initial learning rate was set at 1e-2. During the training, the learning rate gradually decreased with EPOchs to speed up the fitting of loss values. By observing the training progress through the loss value of the model, the training process of the medicine bottle state detection model is shown in Fig. 6.

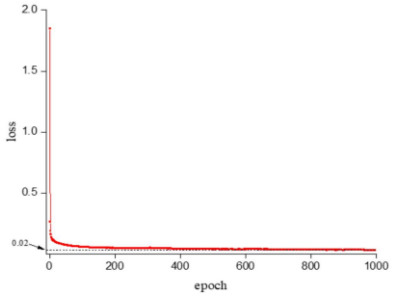

Fig. 6. Loss curve of bottle condition detection model.

Using coco data centrality weight as pre-training weight, the initial loss value of the model is 2.75, and after 14 iterations, the loss value drops below 0.1. Later, with the increase of iterations, the loss value slowly declines, and after 160 iterations, the loss value drops to 0.04. When the model is iterated to 1000 times, the loss value is stable at about 0.02, and the training of the medicine bottle state detection model is completed. The model parameters after the 1000th iteration were taken as the final model parameters, and the drug bottle state detection model was obtained.

Part of the test results of the drug bottle state detection model are shown in Fig. 7. Fig. 7 shows the detection effect of different detection objects on the model. The blue box indicates that the model detects that the medicine bottle is in a disorderly place, the green box indicates that the model detects that the medicine bottle label is off, and the red box indicates that the model detects that the medicine bottle is normal. The confidence degree of the model to the test results is marked on the detection box. In order to evaluate the detection performance of the drug bottle state detection model, the YOLO-v4tiny model and the improved YOLO-v4-tiny model were evaluated on the drug bottle state verification set. The PR curves of the three categories of Nor-drug, Misdrug and Mix-drug on different models are shown in Fig. 8.

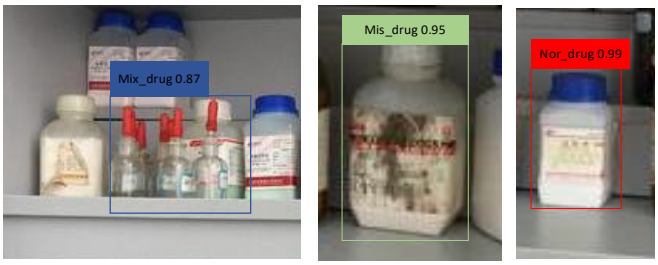

Fig. 7. The test result of drug bottle state test model.

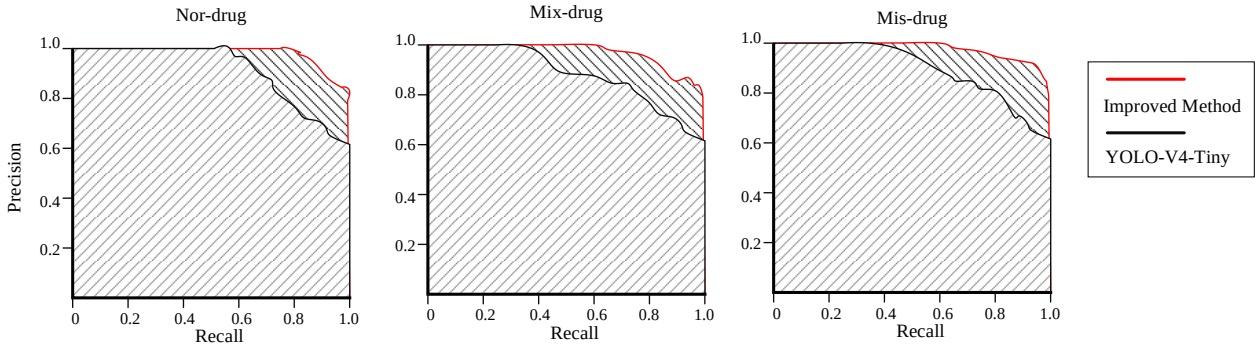

Fig. 8. PR curves of different detection categories on the drug bottle state detection model.

It can be seen from the PR curves of the model that the improved model covers the PR curves of the YOLO-v4-tiny model for the three detection categories of Nor-drug, Mix-drug and Mis-drug, indicating that the model with improved attention mechanism has better detection performance. The AP values of the drug bottle status detection model in various categories and the average detection accuracy of the model are shown in Table VI.

TABLE VI THE AP VALUE OF THE MEDICINE BOTTLE STATE DETECTION MODEL

Detection category	YOLO-v4-tiny	Improved model 1
Nor-drug	84.35%	93.06%
Mis-drug	79.01%	90.72%
Mix-drug	89.42%	95.31%
MAP	84.26%	93.03%

A target detection model must not only accurately identify the target's location and classify the target correctly but also perform detection quickly to meet real-time processing requirements. Table VII presents the FPS (Frames Per Second) results of the bottle status detection model across different categories.

TABLE VII FPS OF THE BOTTLE STATUS DETECTION MODEL FOR DIFFERENT CATEGORIES

Detection Category	FPS (Frames)	Processing Speed per Frame (s)
Nor-drug	272	0.0036
Mis-drug	299	0.0033
Mix-drug	2212	0.0004

As shown in the Table VII, the model achieves an FPS of 272 for "Nor-drug," with each image taking only 0.0036 seconds to process. For "Mis-drug," the FPS is 299, with a processing time of 0.0033 seconds per image. The "Mix-drug" category achieves an FPS of 2212, with a processing time of just 0.0004 seconds per image. These results demonstrate that the bottle status detection model, improved with the attention mechanism, can achieve rapid detection of bottle statuses, meeting real-time processing requirements.

D. Analysis of Results of Unsafe Factors Detection Model in Laboratory

The unsafe factor detection model in the laboratory trained 1000 EPOchs, and the initial learning rate was set at 1e-2, which gradually decreased with the number of iterations. The variation of model loss values with the number of iterations is shown in Fig. 9. The initial loss value of the model was 1.85, and when the model iterated to the 18th epoch, the loss value decreased to 0.09, and then the loss value decreased slowly, and at the 1000th epoch, the loss value decreased to 0.03, and the model loss value tended to be stable. The detection model of unsafe factors in laboratory was obtained.

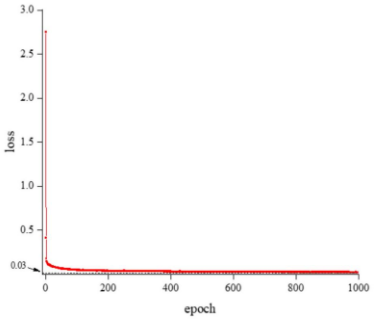

Fig. 9. Loss curve of laboratory unsafe factors detection model.

Part of the image detection results of the unsafe factor detection model in the laboratory are shown in Fig. 10, and the confidence degree of the detection target is shown in Table VIII. As can be seen from above figures and tables, the unsafe factors detection model in the laboratory can accurately select the target to be measured, and has a high degree of confidence in the detection results. It shows that the model can basically realize the detection of not wearing safety protective equipment, smoking and open flame.

Fig. 10. Test results of unsafe factors detection model in laboratory.

Testing the YOLO-v4-tiny model and the improved YOLO-v4tiny model on the laboratory Unsafe Factor validation set, The PR curves of the improved YOLO-v4-tiny laboratory unsafe factor detection model and the original YOLO-v4-tiny model in the four categories of Normal, Fault, Smoke and Fire are shown in Fig. 11.

It can be seen from the PR curves of various types of unsafe factors detection models in the laboratory that the PR curves of the model constructed in this paper wrap the original YOLO-v4-tiny model and have better performance in the unsafe factors detection task. The model showed excellent detection accuracy of Normal, Fault and Smoke in the whole recall rate, which basically reached more than 95%, indicating that the model had high detection performance for the three detection categories. However, it can be seen from the PR curve of the improved model for the detection category Fire that the model's detection performance of open flame needs to be improved, and the model's performance can be improved by increasing the number of training iterations. The AP values of the unsafe factors detection model in the laboratory and the average detection accuracy of the model are shown follow.

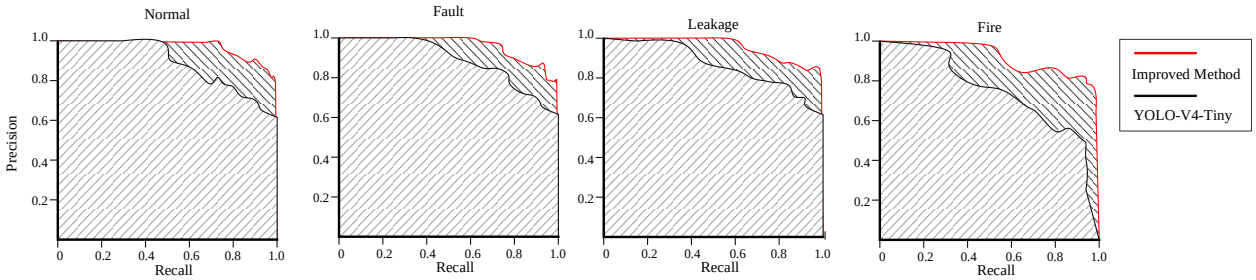

Fig. 11. PR curve of unsafe factors detection model in laboratory.

TABLE VIII AP VALUES OF UNSAFE FACTORS DETECTION MODEL IN LABORATORY IN VARIOUS CATEGORIES

Target class	Normal	Fault	Smoke	Fire	MAP
AP	97.40%	90.14%	96.80%	68.95%	88.32%

The AP values of the unsafe factors detection model in the laboratory reached 97.40%, 90.14% and 96.80% for Normal, Fault and Smoke, respectively, indicating that the model has a good detection effect on these three categories. The AP value of the model for open flame (Fire) reached 68.95%, and the average detection accuracy of the model reached 88.32%. The model basically meets the requirement of detecting unsafe factors in laboratory. The FPS values for each category detected by the unsafe factor detection model in the laboratory are shown in Table IX.

TABLE IX FPS VALUES OF THE UNSAFE FACTOR DETECTION MODEL FOR EACH CATEGORY IN THE LABORATORY

Detection Category	FPS (Frames)	Processing Speed per Frame (s)
Normal	1110	0.0009
Fault	846	0.0012
Smoke	116	0.0086
Fire	2937	0.0003

As shown in the table, the model achieves an FPS of 1110 for the "Normal" category, requiring only 0.0009 seconds to process each image. For the "Fault" category, the FPS is 846, with a processing time of 0.0012 seconds per image. The "Smoke" category has an FPS of 116, with each image taking 0.0086 seconds to process. Lastly, the "Fire" category achieves an FPS of 2937, with a processing time of only 0.0003 seconds per image. The model meets the real-time processing requirements.

V. CONCLUSION

This study addresses the critical need for safety management in environments where hazardous chemicals are stored and used, such as laboratories. By leveraging safety engineering principles, a highly efficient model for identifying unsafe factors was developed, significantly enhancing the intelligence of laboratory safety monitoring. The study employed a lightweight YOLOv4-tiny algorithm, optimized with techniques such as CIOU for more stable bounding box regression and the integration of attention mechanism modules, to improve the model's performance in detecting unsafe factors. In addition, experiments were conducted to demonstrate the effectiveness of the improved algorithm. In summary, the main contributions are as follows:

1) Proposing and optimizing the YOLOv4-tiny algorithm, making it more suitable for the task of recognizing unsafe factors in laboratories, while balancing lightweight design with high efficiency.

2) Developing a dataset for unsafe laboratory conditions, providing crucial foundational data for future related research.

3) Validating the potential of deep learning in laboratory safety monitoring, laying a solid technical foundation for the development of intelligent laboratory safety management systems.

These contributions not only provide effective technical support for chemical laboratory safety monitoring but also offer valuable experience and data for future research and development in related technologies, further advancing the intelligence of laboratory safety management.

ACKNOWLEDGMENT

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

REFERENCES

[1] Ren Y, Dong J, He J, et al. A novel six-dimensional digital twin model for data management and its application in roll forming[J]. Advanced Engineering Informatics, 2024, 61: 102555.

[2] Xu M, Liu S, Shen H, et al. Process-oriented unstable state monitoring and strategy recommendation for burr suppression of weak rigid drilling system driven by digital twin[J]. The International Journal of Advanced Manufacturing Technology, 2022: 1-17.

[3] Dalal N, Triggs B. Histograms of oriented gradients for human detection[C]. IEEE Computer Society Conference on Computer Vision & Pattern Recognition, 2005.

[4] Felzenszwalb P F, Mcallester D A, Ramanan D. A discriminatively trained, multiscale, deformable part model[C]. 2008 IEEE Conference on Computer Vision and Pattern Recognition, 2008: 1-8.

[5] Felzenszwalb P F, Girshick R B, Mcallester D A. Cascade object detection with deformable part models[C]. 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2010: 41-48.

[6] Girshick R B, Felzenszwalb P F, Mcallester D A. Object detection with grammar models[J]. Advances in Neural Information Processing Systems, 2011(1): 442-450.

[7] Fukushima K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position[J]. Biological Cybernetics, 1980, 36(4): 193202.

[8] Lecun Y, Boser B, Denker J, et al. Backpropagation applied to handwritten zip code recognition[J]. Neural Computation, 1989, 1(4): 541-551.

[9] Krizhevsky A, Sutskever I, Hinton G. ImageNet classification with deep convolutional neural networks[J]. Advances in Neural Information Processing Systems, 2012, 25(2): 1097-1105.

[10] Zhong Z, Jin L, Xie Z. High performance offline handwritten chinese character recognition using googlenet and directional feature maps[C]. 2015 13th International Conference on Document Analysis and Recognition, 2015: 846-850.

[11] Wu Z, Shen C, Hengel A. Wider or deeper: Revisiting the resnet model for visual recognition[J]. Pattern Recognition, 2016, 90: 119-133.

[12] Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]. IEEE Conference on Computer Vision and Pattern Recognition, 2014: 580-587.

[13] Girshick R. Fast R-CNN[C]. IEEE International Conference on Computer Vision, 2015: 14401448 .

[14] Redmon J, Divvala S, Girshick R, et al. You Only Look Once: Unified, Real-time object detection[C]. IEEE Conference on Computer Vision and Pattern Recognition, 2016: 779-788.

[15] Redmon J, Farhadi A. YOLO9000: better, faster, stronger[C]. IEEE Conference on Computer Vision & Pattern Recognition, 2017: 6517-6525.

[16] Redmon J, Farhadi A. YOLOv3: YOLO-V3: An incremental improvement[C]. IEEE Conference on Computer Vision and Pattern Recognition, 2018:89-95.