

Article

Not peer-reviewed version

Research on a Real-Time Warning System for Unsafe Behaviors in Hydraulic Construction Based on DeepSORT and Improved YOLOv5s

Yongqiang Liu , [Haibin Wu](#)*, Haomin Li

Posted Date: 14 January 2026

doi: 10.20944/preprints202601.0998.v1

Keywords: hydraulic engineering construction; unsafe behavior; YOLOv5s; DeepSORT; multi-object tracking; real-time warning system



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a [Creative Commons CC BY 4.0 license](#), which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

Research on a Real-Time Warning System for Unsafe Behaviors in Hydraulic Construction Based on DeepSORT and Improved YOLOv5s

Yongqiang Liu ¹, Haibin Wu ^{1,*} and Haomin Li ²

¹ College of Water Conservancy and Hydropower Engineering, Hohai University, Nanjing, China 1; 210098

² NARI Group Corporation, State Grid, Nanjing China, 2; 211000

* Correspondence: 231302050005@hhu.edu.cn

Abstract

The construction environment of hydraulic engineering is complex, while traditional safety monitoring methods suffer from low efficiency and delayed response. Although static recognition models based on improved YOLOv5s have enhanced detection accuracy, they still cannot assess behavioral persistence and struggle to achieve proactive early warning. To address this, this study integrates the improved YOLOv5s with the DeepSORT algorithm to construct an integrated real-time "detection-tracking-warning" system. The system utilizes DeepSORT to achieve stable personnel tracking in complex scenarios and triggers dynamic warnings based on spatiotemporal behavioral logic. A desktop prototype system was developed using PyQt5/PySide6. Experimental results show that the system achieves a Multiple Object Tracking Accuracy (MOTA) of 86.2% in multi-object occlusion scenarios; the accuracy of unsafe behavior warning exceeds 95%, with an average delay of less than 1.5 seconds. This research accomplishes a transition from passive recognition to proactive warning, providing an intelligent solution for safety management in hydraulic construction.

Keywords: hydraulic engineering construction; unsafe behavior; YOLOv5s; DeepSORT; multi-object tracking; real-time warning system

1. Introduction

As a critical pillar of national infrastructure development, hydraulic engineering construction often involves complex and high-risk working environments such as heights, proximity to water, and deep foundation pits. Unsafe behaviors of construction personnel have become a major contributing factor to safety incidents [1]. Traditional safety management models primarily rely on on-site safety inspections and post-event video reviews, which suffer from inherent limitations such as limited supervision scope, high labor costs, and delayed risk response [2,3]. This makes real-time dynamic monitoring across the entire construction process and all scenarios difficult to achieve. In recent years, with the rapid advancement of artificial intelligence and computer vision technologies, intelligent vision-based monitoring systems utilizing deep learning have emerged as a significant direction for improving construction safety management [4,5]. Such systems can automatically identify unsafe behaviors of construction personnel through video data and provide real-time warnings, thereby promoting a paradigm shift in safety management from "post-event traceability" to proactive prevention involving "in-process intervention" and "pre-event prevention." This holds significant theoretical and practical importance for ensuring personnel safety [4], reducing accident losses, and fostering smart construction sites [6].

Regarding the intelligent recognition and warning of unsafe construction behaviors, scholars worldwide have conducted extensive exploration and achieved phased results. However, several limitations persist. As summarized in Table 1, most existing work focuses on a single aspect: either utilizing object detection algorithms (e.g., Faster R-CNN, YOLO series) at the behavior recognition

level for static discrimination of personal protective equipment (PPE) such as safety helmets and reflective vests [7,8]; or introducing algorithms like SORT at the tracking level for area intrusion warnings. However, static recognition methods cannot assess behavioral persistence. Algorithms like SORT, due to their lack of appearance feature utilization, are prone to frequent identity (ID) switching in complex scenarios. Our team's prior research improved static detection accuracy by enhancing the YOLOv5s model [1], but it did not address the core issue of dynamic warning: "determining the continuous state of a target." The DeepSORT algorithm significantly enhances the robustness of multi-object tracking by fusing motion and appearance information. However [8,9], its application in hydraulic construction scenarios remains insufficiently explored, and there is a lack of end-to-end systems that deeply integrate high-precision detection models with practical engineering deployment in mind [10].

Table 1. Comparison of Related Research on Construction Safety Monitoring.

Detection Model	Tracking Algorithm	Dynamic Warning?	Application Scenario	Key Metrics
Faster R-CNN	None	No	General Construction	mAP: 85.3%
YOLOv3	SORT	Yes (Area Intrusion)	Construction Site	MOTA: ~78%
SSD	DeepSORT	Partial	Traffic Monitoring	IDF1: 82.1%
Improved YOLOv5s	DeepSORT	Yes (Behavior Persistence + Area Intrusion)	Hydraulic Construction	MOTA: 86.2%, Warning Accuracy: 95.2%

Note: This work achieves deep integration of high-precision detection, robust tracking, and dynamic warning, with targeted optimization and system integration for hydraulic construction scenes (small objects, severe occlusion, variable lighting) [11].

Hydraulic construction scenarios present unique challenges: (1) Complex Environment: Open-air operations are significantly affected by lighting and weather, exhibiting day/night and sunny/rainy variations; (2) Target Characteristics: Personnel often wear similar work uniforms, with small targets (distant personnel), dense crowds, and severe occlusion; (3) Safety Regulations: There are clear requirements for behavioral persistence (e.g., continuously not wearing a safety helmet constitutes a risk). These challenges impose higher demands on the algorithm's robustness, real-time performance, and the rationality of warning rules [13].

In summary, to achieve a systematic leap from "static recognition" to "dynamic warning," this study aims to tackle two core problems: (1) Stable tracking in complex scenes: Achieving high-precision, consistent tracking of multiple targets within occlusion-prone, dense, and variable hydraulic construction video streams [14]; (2) Dynamic warning logic compliant with regulations: Transforming discrete detection events into continuous warning decisions based on spatiotemporal trajectories that align with safety protocols. The main contributions include: (1) Proposing a deep integration framework combining improved YOLOv5s and DeepSORT, with detailed elaboration on key parameter settings and optimization of the feature extraction network; (2) Developing a desktop integrated warning system based on PyQt5/PySide6, realizing algorithm engineering; (3) Conducting comprehensive validation on real hydraulic engineering data, evaluating not only tracking performance but also, through detailed comparative experiments and real-time analysis, demonstrating the system's advancement and practical potential [15,16].

2. Related Theoretical and Technical Foundations

(1) Overview of the Multi-Object Tracking Algorithm DeepSORT

DeepSORT (Deep Simple Online and Realtime Tracking) extends the SORT algorithm. Its core innovation lies in leveraging deep appearance features to enhance the robustness of data association [8]. The algorithm primarily consists of the following components [17]:

Kalman Filter

The Kalman filter in DeepSORT predicts a target's state (e.g., position, velocity) in the next frame. Its core comprises the state equation, observation equation, and the prediction-update recursive process [14].

State Equation

The state equation describes the temporal evolution of the target state using a linear model :

$$x_k = Fx_{k-1} + w_{k-1} \quad (1)$$

where x_k is the state vector (containing position and velocity); F is the state transition matrix; w_{k-1} is the process noise.

Observation Equation

The observation equation expresses the mapping from the state to the observation.

$$z_k = Hx_{k-1} + v_k \quad (2)$$

where z_k is the observation vector ; H is the observation matrix; v_k is the observation noise.

Prediction Step

(a) State Prediction

$$\hat{x}_{k|k-1}^- = F_k \hat{x}_{k-1|k-1}^- + B_k u_k \quad (3)$$

where $\hat{x}_{k|k-1}^-$ is the predicted state estimate at time k ; F_k is the state transition matrix ; $\hat{x}_{k-1|k-1}^-$ is the previous state estimate at time $(k-1)$; B_k is the control input matrix incorporating the influence of the control vector u_k into the state estimate.

(b) Covariance Prediction

$$P_{k|k-1}^- = F_k P_{k-1|k-1}^- F_k^T + Q_k \quad (4)$$

where $P_{k|k-1}^-$ is the covariance matrix of the predicted state estimate; F_k is the state transition matrix; $P_{k-1|k-1}^-$: is the previous covariance matrix; and Q_k is the covariance matrix of the process noise. [18,19]

Update Step

(a) Kalman Gain

$$K_k = \frac{P_{k|k-1}^- H^T}{H P_{k|k-1}^- H^T + R_k} \quad (5)$$

(b) State Update

$$\hat{x}_{k|k} = \hat{x}_{k|k-1}^- + K_k (z_k - H \hat{x}_{k|k-1}^-) \quad (6)$$

(c) Covariance Update

$$P_{k|k} = (I - K_k H) P_{k|k-1}^- \quad (7)$$

where K_k is the Kalman gain, determining the weight of new observation data relative to the prediction; $P_{k|k}$ is the covariance matrix of the updated state estimate; H is the observation matrix; z_k is the actual observation at time k ; R_k is the covariance matrix of the observation noise; and I : is the identity matrix.

DeepSORT achieves relatively high robustness in multi-object tracking under complex scenarios such as occlusion by fusing the Kalman filter with deep learning features [21].

Hungarian Matching Algorithm

The Hungarian algorithm is a combinatorial optimization method commonly used to solve optimal matching problems in bipartite graphs. In multi-object tracking, it matches currently detected targets with existing tracks. Its basic process involves constructing a cost matrix representing the matching cost between detections and tracks, and then finding the globally optimal matching combination through steps like row/column reduction. In DeepSORT, this algorithm enables efficient and optimal association, ensuring tracking accuracy and stability.

(2) System Development Related Technologies

To realize the engineering application, this study employs PyQt5/PySide6 as the graphical user interface (GUI) development framework. PyQt5/PySide6 are Python bindings for the Qt library, offering cross-platform compatibility, a rich set of widgets, and efficient design. Combined with the Qt Designer visual tool, it enables rapid construction of desktop applications containing modules like video display and control panels. The signal and slot mechanism facilitates efficient interaction between the front-end interface and back-end algorithm logic, meeting the system's real-time requirements.

3. Design of the Unsafe Behavior Warning Method Integrating DeepSORT

3.1. Analysis of Warning Scenarios and Principles

Based on hydraulic engineering characteristics, this system primarily designs warnings for two scenarios: (1) Missing Personal Protective Equipment (PPE): Personnel not wearing safety helmets or reflective vests; (2) Dangerous Area Intrusion: Unauthorized personnel entering predefined hazardous areas (e.g., foundation pits, edges, crane zones).

Warnings adhere to: Accuracy: Minimizing false positives/negatives; Real-time Performance: Meeting engineering monitoring speed requirements; Compliance: Following safety regulations. For instance, for "not wearing a safety helmet," an alarm is not triggered immediately. According to the Technical Code for Safety of High-altitude Operation in Building Construction (JGJ80-2016), brief exposure (<3 seconds) can be considered accidental, while exceeding 5 seconds constitutes a clear hazard. Considering site video analysis and the need to balance sensitivity with resistance to brief interference (e.g., momentary occlusion), this study sets the alarm threshold to 7 consecutive seconds, ensuring both rigor and practicality.

3.2. Warning Model Construction: YOLOv5-DeepSORT Integrated Framework

The proposed model achieves deep integration of the improved YOLOv5s detection module and the DeepSORT tracking module, forming an end-to-end real-time "detection-tracking-warning" pipeline. The overall workflow is shown in Figure 1, with core mechanisms detailed below.

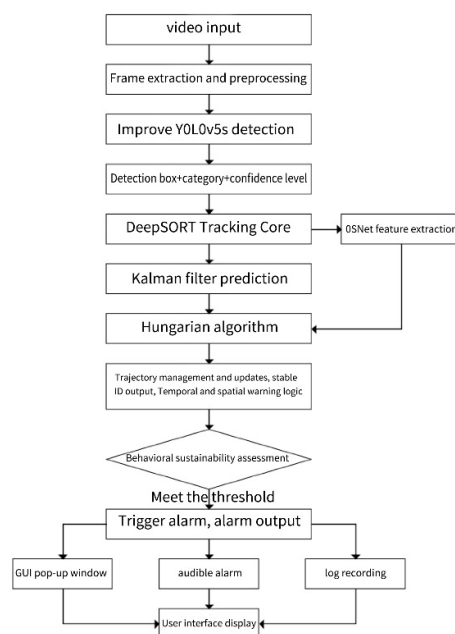


Figure 1. Architecture of the Improved YOLOv5-DeepSORT Integrated Early Warning System.

(1) Video Input and Image Preprocessing

The system supports multiple video source inputs, including real-time cameras and local surveillance recordings. Input video frames first undergo preprocessing such as scale normalization and color space conversion to adapt to the input requirements of the deep learning model.

(2) Target Detection Based on Improved YOLOv5s

Preprocessed frames are fed into the improved YOLOv5s model, which incorporates a Coordinate Attention (CA) mechanism and a Focal-EIoU loss function, enhancing detection accuracy for PPE in complex scenes. The model outputs bounding box, class label, and confidence for each target. The detection confidence threshold is set to 0.5.

(3) Deep Feature Extraction and Motion Prediction

Appearance Feature Extraction: For each detected personnel target, its corresponding image region is cropped and input into an OSNet network to extract a 128-dimensional appearance feature vector. This network was pre-trained on the Market-1501 person re-identification dataset and fine-tuned using self-collected images of hydraulic construction personnel, thereby enabling better discrimination between different individuals wearing similar work uniforms.

Motion State Prediction: Simultaneously, the system uses the Kalman filter to predict the position and velocity of existing track targets in the next frame. Filter parameters are set according to the typical movement patterns of personnel in the scene, with the process noise covariance matrix $Q_k = \text{diag}(1, 1, 10, 10)$ and the observation noise covariance matrix $R_k = \text{diag}(1, 1, 1, 1)$ [22].

(4) Data Association and Track Lifecycle Management

Association Matching: The Hungarian algorithm is employed for data association, matching the detection results of the current frame with the tracks predicted by the Kalman filter. The matching cost function considers both the Intersection over Union (IoU) of bounding boxes and the cosine similarity of the appearance features extracted by OSNet. The IoU matching threshold is set to 0.3.

Track Management: Successfully matched tracks are updated with the current detection information. Unmatched detections are initialized as new tracks but require successful matching in 3 consecutive subsequent frames for final confirmation, in order to filter out transient false detections. For tracks that have lost their match, the system maintains a buffer period of 30 frames to handle brief occlusions. Tracks exceeding this period are considered to have left the scene and are deleted [12].

(5) Dynamic Warning Based on Spatiotemporal Logic

The system maintains a spatiotemporal status stack for each active track. When a specific unsafe behavior (e.g., "not wearing a safety helmet") is identified, continuous timing for that target begins. If this behavioral state remains continuous beyond a preset threshold (e.g., 7 seconds), a dynamic warning is triggered. This threshold is determined based on the regulations regarding continuous exposure risk in the *Technical Code for Safety of High-altitude Operation in Building Construction (JGJ80-2016)* and combined with practical on-site monitoring requirements, aiming to reduce false alarms caused by momentary occlusion or posture changes.

(6) Multimodal Alert and Interactive Interface

Upon warning trigger, the system provides multimodal output through the PyQt5/PySide6-based desktop application: the interface pops up a visual warning box, plays an alert sound effect, and logs event details (time, target ID, violation type, screenshot) to a log database. The main interface simultaneously renders the real-time video stream, overlaying target detection boxes, tracking trajectories, and warning statuses, providing an intuitive monitoring view for safety managers [23].

This integrated framework, through the tight coupling of high-precision detection, robust multi-object tracking, and regulation-compliant warning logic, achieves an automated closed-loop process from perception to warning for unsafe behaviors in hydraulic construction scenarios. It provides a practical and feasible technical solution for enhancing the intelligence level of safety management.

In summary, the specific early warning algorithm is implemented based on the DeepSORT algorithm according to the steps above, with the code implementation as shown in Figure 2 below.

```

1 import argparse
2 import logging
3 import math
4 import os
5 import random
6 import time
7 from copy import deepcopy
8 from pathlib import Path
9 from threading import Thread
10
11 import numpy as np
12 import torch.distributed as dist
13 import torch.nn as nn
14 import torch.nn.functional as F
15 import torch.optim as optim
16 import torch.optim.lr_scheduler as lr_scheduler
17 import torch.utils.data
18 import yaml
19
20 from torch.cuda import amp
21 from torch.nn.parallel import DistributedDataParallel as DDP
22 from torch.utils.tensorboard import SummaryWriter
23 from tqdm import tqdm
24
25 import time # import test.py to get amp after each epoch
26 from models.experimental import attempt_load
27 from models.yolo import Model
28
29 from utils.autoanchor import check_anchors
30 from utils.datasets import create_dataloader
31 from utils.general import labels_to_class_weights, increment_path, labels_to_image_weights, init_seeds, \
32     fitness, strip_optimizer, get_latest_run, check_dataset, check_file, check_git_status, check_img_size, \
33     check_requirements, print_mutation, set_logging, one_cycle, colorstr
34 from utils.google_utils import attempt_download
35 from utils.loss import ComputeLoss
36 from utils.plots import plot_images, plot_labels, plot_results, plot_evolution
37 from utils.torch_utils import ModelEMA, select_device, intersect_dicts, torch_distributed_zero_first, is_parallel
38 from utils.wandb_logging.wandb_utils import WandbLogger, check_wandb_resume
39
40 logger = logging.getLogger(__name__)

```

Figure 2. Early warning algorithm implementation.

4. Implementation of the Unsafe Behavior Recognition and Warning System

3.1. System Design Objectives and Architecture

The design objectives of this system are: to provide a user-friendly desktop application with a modular design. The architecture mainly includes a Data Input Module, a Data Processing Module, an Unsafe Behavior Recognition Module, an Unsafe Behavior Warning Module, and a Data Storage Module. The specific details of the architecture are shown in Figure 3 below.

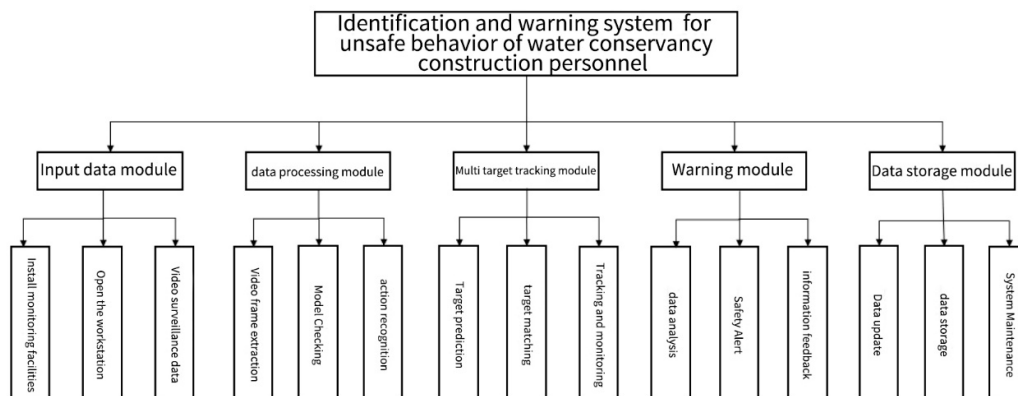


Figure 3. The main modules of the system.

3.2. System Interface Design and Functional Modules

The main system interface designed using Qt Designer is shown in Figure 4, with its core functional areas including:

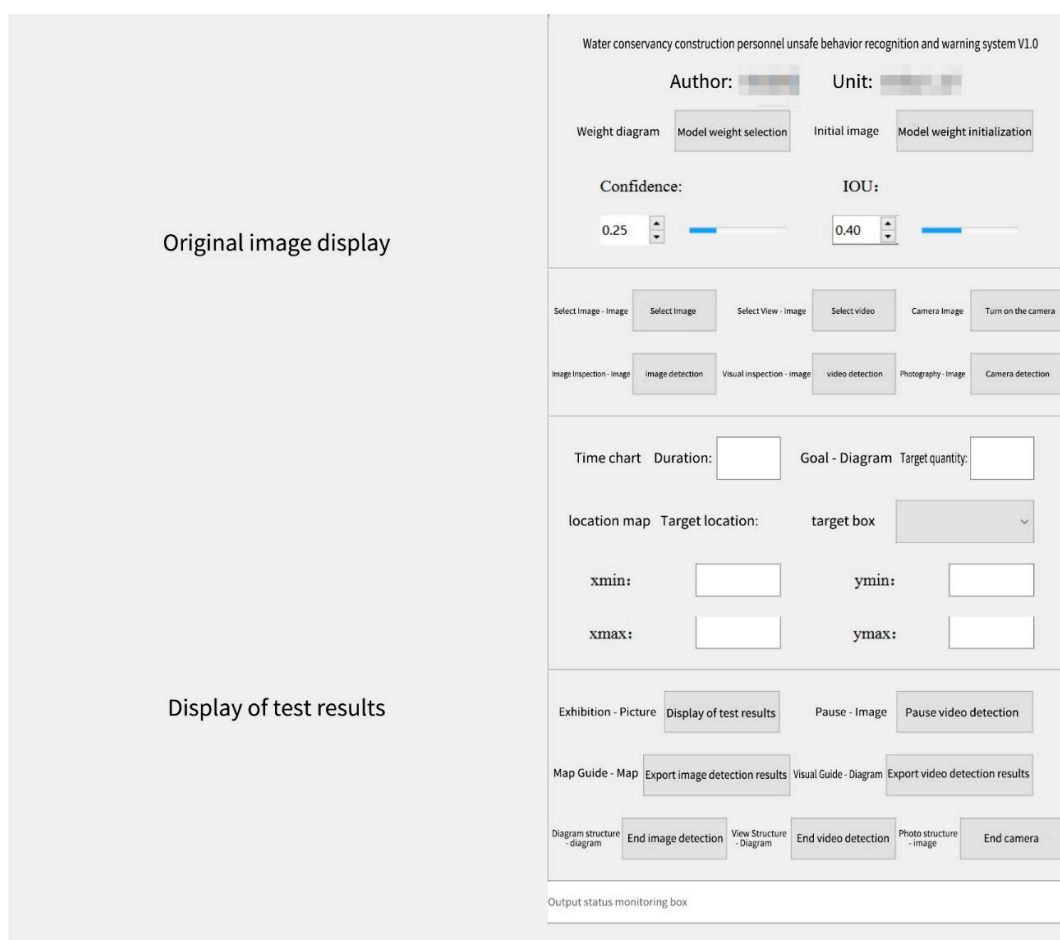


Figure 4. Qt Designer interface implementation.

The system employs multi-threading technology to ensure a smooth and responsive UI interface, while the backend continuously performs video decoding, model inference, and warning assessment.

5. Empirical Research and System Validation

5.1. Experimental Environment and Data Preparation

The experimental hardware environment was as follows: Intel i7-13700KF processor, NVIDIA RTX 4060 Ti graphics card (16GB VRAM), and 64GB RAM. The software environment consisted of Ubuntu 22.04, PyTorch 1.12.1, and Python 3.8. The test data was sourced from six months of construction site surveillance video of a sluice reconstruction project in Jiangsu Province. Detailed information about the dataset is presented in Table 2 below [24].

Table 2. Composition of the Experimental Dataset.

Item	Description
Total Video Duration	Approximately 120 hours
Annotated Frames	15,000 frames (extracted)
Person Bounding Boxes	Approximately 45,000
Scene Distribution	Daytime (70%), Night (20%), Dawn/Dusk (10%)
Challenge Types	Occlusion (35%), Dense Crowds (25%), Lighting Variation (40%)
Publicly Available?	Not yet publicly available; consideration will be given to providing partially anonymized data after publication

5.2. Warning Model Performance Testing

To comprehensively evaluate system performance, a comparative experiment on tracking performance and a warning efficacy test were designed.

(1) Comparative Experiment on Tracking Performance

On the same test set, the system proposed in this paper (Improved YOLOv5s + DeepSORT) was compared with various mainstream combinations. The results are shown in Table 3.

Table 3. Comparison of Multi-Object Tracking Performance.

Method	MOTA (%) ↑	IDF1 (%) ↑	FPS ↑
YOLOv5s + SORT	79.5	80.2	62
YOLOv5s + ByteTrack	84.1	86.5	58
YOLOv8m + DeepSORT	85.8	88.0	40
FairMOT	83.3	85.7	30
Ours (Improved YOLOv5s+DeepSORT)	86.2	88.7	41

The analysis indicates that the proposed method achieves optimal results in both MOTA and IDF1, demonstrating the effectiveness of fusing the improved YOLOv5s with DeepSORT. Although the FPS is slightly lower than that of the pure motion-based model SORT, the processing speed of 41 FPS far exceeds the 25-30 FPS requirement for real-time monitoring, achieving a balance between accuracy and speed.

(2) Warning Efficacy Test

Testing was conducted using 100 independent video clips containing violations, which triggered a total of 312 warnings. The system correctly issued 297 warnings, with 10 false positives and 5 false negatives. The calculated warning accuracy reached 95.2%. The warning delay, measured from the moment the behavior continuously met the threshold until the system interface popped up the alarm, averaged 1.3 seconds, meeting the requirement for real-time intervention [25].

5.3. Overall System Operation Verification

(1) Loading the Mode

Start the system; the system startup interface is shown in Figure 5 below.



Figure 5. System boot screen.

Load the trained model weight file (best.pt) into the system. Click the "Model Weight Selection" button to load the local model best.pt into the system. The model loading interface is shown in Figure 6 below.

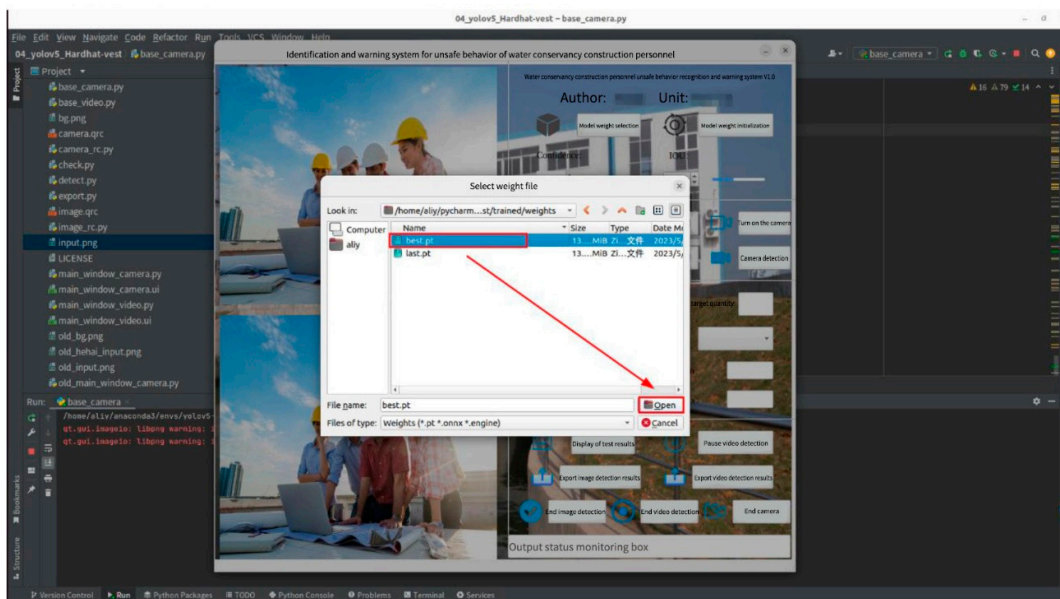


Figure 6. The system loads the model interface.

After loading, click the "Initialize Model Weights" button on the right. The model loading is then completed, and initialization is successful, as shown in Figure 7 below.

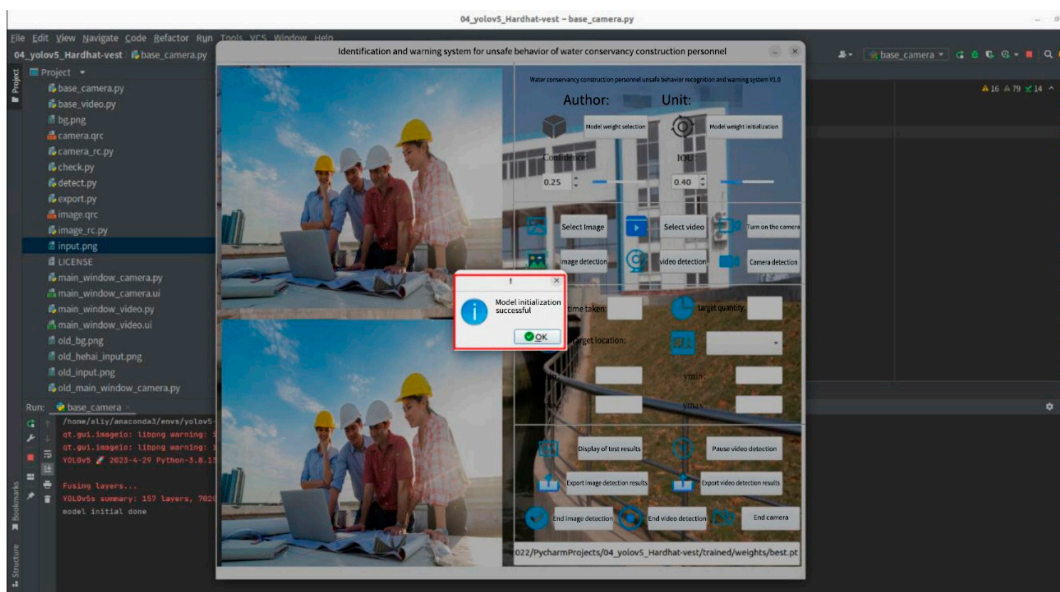


Figure 7. The system model initialization success page.

(2) Selecting Data

Based on practical requirements, this system has been designed and implemented to support three data sources: image data, video data, and camera data. Taking image data selection as an example, the system is validated as follows. The data selection interface is shown in Figure 8 below.

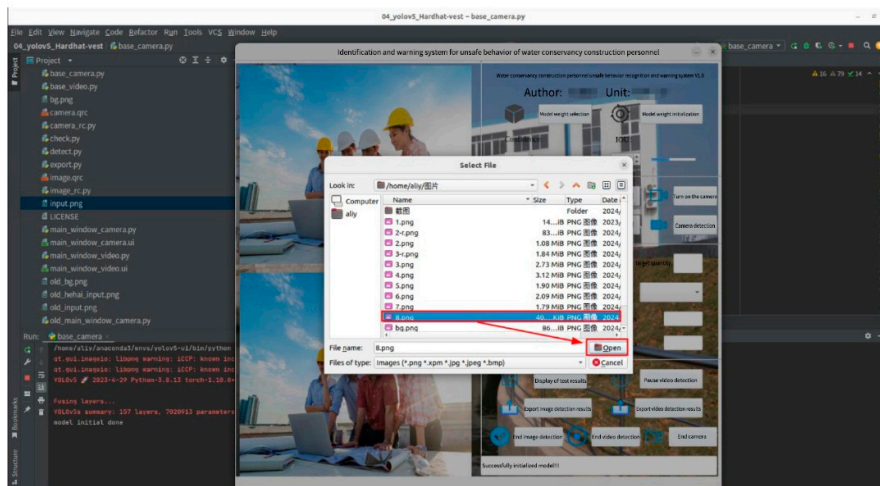


Figure 8. The system selects the image interface.

By clicking the "open" button, image files can be loaded, and the detection of unsafe behaviors among construction personnel can be executed. The operational process for video files or real-time camera streams is similar.

(3) Verification of Unsafe Behavior Recognition

After loading an image, the "Image Detection" function can be selected to invoke the model for unsafe behavior recognition. Upon completion of detection, the system will prompt that detection was successful. Subsequently, specific detection results can be viewed in the lower-left section of the interface, as shown in Figure 9.

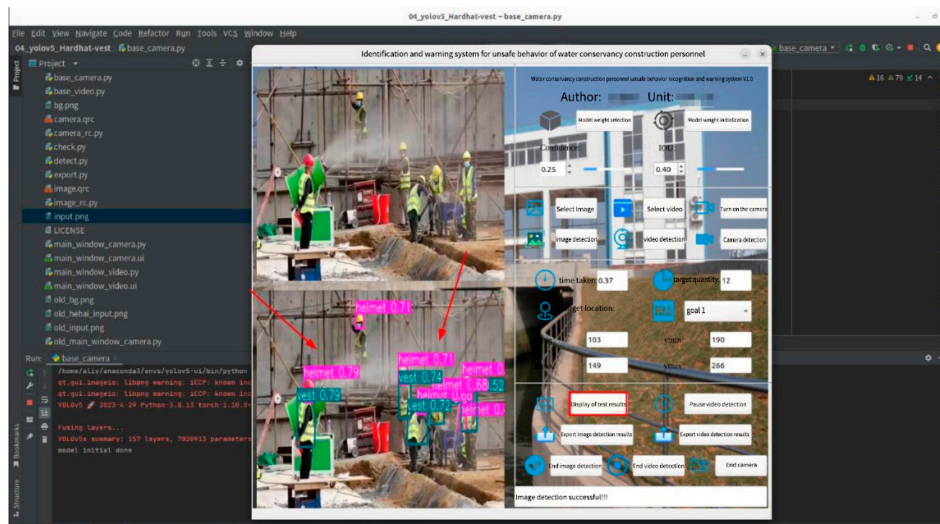


Figure 9. The system test results are displayed.

From the above results, it can be concluded that this system is capable of identifying and detecting unsafe behaviors of construction personnel. To test the model's performance from multiple aspects and dimensions, experiments will be conducted under two special conditions: testing with ordinary hats and under rainy/overcast weather.

(a) Ordinary Hat Detection Experiment

Safety helmets and ordinary hats are highly similar in appearance. In construction safety detection, the model must accurately distinguish between them, rather than making a simple binary judgment of "wearing" or "not wearing." Mistaking an ordinary hat for a safety helmet would lead to

false negatives in the system, reducing warning reliability. Based on the collected construction site data, the experimental verification results are shown in Figure 10 below.



Figure 10. Common hat test results.

The detection results indicate that the construction worker in the experiment is wearing an ordinary baseball cap, not a safety helmet. The model trained based on the improved YOLOv5s target detection algorithm can accurately identify the ordinary hat as head (indicating no safety helmet), which meets the safety inspection requirements for engineering construction sites. This demonstrates that the model possesses good generalization ability.

(b) Nighttime Construction Detection Experiment

Hydraulic engineering often requires nighttime construction due to factors such as project schedules. Nighttime lighting conditions differ significantly from daytime. Low illumination and non-uniform artificial light sources can easily reduce target visibility and cause color distortion, posing challenges to the detection model. Therefore, this study uses nighttime construction site data for testing to evaluate the model's robustness under low-light conditions. The detection results are shown in Figure 12 below.



Figure 11. Night-time construction inspection results.

The experiments demonstrate that the model can still accurately identify safety helmets and reflective vests in nighttime environments. This indicates its good adaptability to varying illumination conditions, enabling support for all-weather monitoring of unsafe behaviors and contributing to the improvement of safety levels during nighttime construction.

(4) Verification of Unsafe Behavior Warning

Accurately and timely identifying unsafe behaviors of construction personnel is crucial for construction safety management. This paper, based on the DeepSORT target tracking algorithm, achieves continuous tracking of construction personnel and issues timely alarms according to the designed warning mechanism to prevent safety incidents. At the code level, the system integrates the aforementioned algorithm and warning logic: taking the example of not wearing a safety helmet, if the same target is continuously detected as not wearing a safety helmet for 7 consecutive seconds, the system will pop up an alarm prompt on the interface and log the warning information in the background. The specific warning interface is shown in Figure 12.

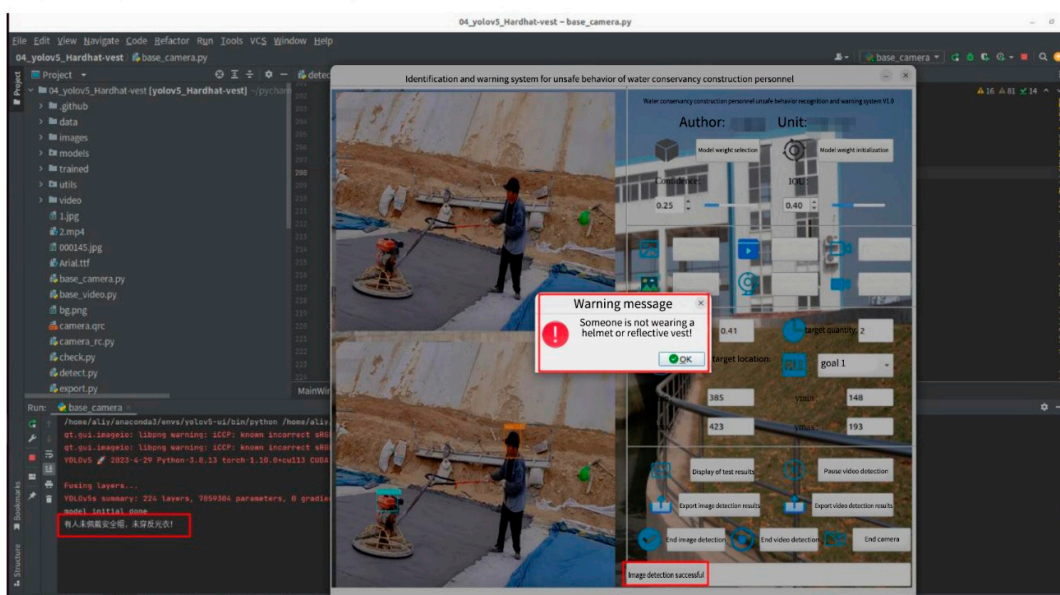


Figure 12. The system warning interface is displayed.

6. Discussion

Addressing the need for intelligent safety management in hydraulic construction, this study successfully constructed a real-time visual monitoring system integrating "detection-tracking-warning." By deeply fusing the improved YOLOv5s with the DeepSORT algorithm and designing the warning logic based on safety regulations, the system effectively resolved the challenges of stable personnel tracking and persistent behavior judgment in complex scenarios. Empirical research demonstrates that the system performs excellently in terms of multi-object tracking accuracy (MOTA 86.2%) and dynamic warning accuracy (95.2%), while its processing speed meets real-time requirements. This study preliminarily achieved the transition from a laboratory algorithm model to an engineering prototype system, providing a solution with practical implementation potential for intelligent safety monitoring in hydraulic construction.

This study still has limitations. Future work will focus on the following directions:

Model Lightweighting and Edge Deployment: We plan to use TensorRT for model acceleration and quantization, and port it to edge devices such as Jetson AGX Xavier to achieve front-end intelligent analysis.

Expansion of Complex Behavior Recognition: Integrate lightweight pose estimation models (e.g., YOLO – Pose) to recognize more complex unsafe behaviors such as climbing and falling.

Multimodal Fusion Warning: Plan to fuse UWB high-precision positioning data to construct a centimeter-level accuracy spatial intrusion warning mechanism, enhancing system reliability.

Closed-loop System Management: Link the warning system with on-site audible/visual alarms and project management platforms to achieve a closed-loop safety management process from “perception-warning-response-feedback.”

Author Contributions: L. Y. and W. H. Writing – original draft; L. Y. and W. H. writing—review and editing; L. H. – visualisation.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Liu, Y.; Wang, P.; Li, H. An Improved YOLOv5s-Based Algorithm for Unsafe Behavior Detection of Construction Workers in Construction Scenarios. *Appl. Sci.* 2025, 15, 1853.
2. Zhang, M.; Shi, R.; Yang, Z. A Critical Review of Vision-Based Occupational Health and Safety Monitoring of Construction Site Workers. *Saf. Sci.* 2020, 126, 104658.
3. Yu, Y.H.; Zhou, C.D.; Meng, M.H.; et al. An Identification Method for Unsafe Behavior of Elevator Personnel Based on Improved YOLOv5 Algorithm. *Journal of Chongqing University of Science and Technology (Natural Sciences Edition)* 2022, 24(02), 79–83+98.
4. Luo, G.; Wang, Y.; Li, H.; Yang, W.; Lyu, L. Unsafe behavior real-time detection method of intelligent workshop workers based on improved YOLOv5s. *Comput. Integr. Manuf. Syst.* 2024, 30, 1610–1619. [Google Scholar]
5. Kisaeezra; Farooq, M.U.; Bhutto, M.A.; Kazi, A.K. Real-Time Safety Helmet Detection Using Yolov5 at Construction Sites. *Intell. Autom. Soft Comput.* 2023, 36, 911–927. [Google Scholar] [CrossRef]
6. Han, K.; Zeng, X. Deep Learning-Based Workers Safety Helmet Wearing Detection on Construction Sites Using Multi-Scale Features. *IEEE Access* 2022, 10, 718–729. [Google Scholar] [CrossRef]
7. Yang, Y.; Li, D. Lightweight HelmetWearing Detection Algorithm of Improved YOLOv5. *Comput. Eng. Appl.* 2022, 58, 201–207. [Google Scholar]
8. Fang, W.; Ding, L.; Luo, H.; et al. Falls from Heights: A Computer Vision-Based Approach for Safety Harness Detection. *Autom. Constr.* 2018, 91, 53–61.
9. Zhang, S.R.; Liang, B.J.; Ma, C.G.; et al. A Method for Identifying Unsafe Behavior of Construction Workers in Hydraulic Engineering. *Journal of Hydroelectric Engineering* 2023, 42(08), 98–109.
10. Wojke, N.; Bewley, A.; Paulus, D. Simple Online and Realtime Tracking with a Deep Association Metric. In *Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017*; IEEE: Piscataway, NJ, USA, 2017; pp. 3645–3649.
11. Wang, Y.; Huang, Y.; Gu, B.; Cao, S.; Fang, D. Identifying mental fatigue of construction workers using EEG and deep learning. *Autom. Constr.* 2023, 151, 104887. [Google Scholar] [CrossRef]
12. Bewley, A.; Ge, Z.; Ott, L.; et al. Simple Online and Realtime Tracking. In *Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016*; IEEE: Piscataway, NJ, USA, 2016; pp. 3464–3468.
13. Ning, M.; Ma, H.; Wang, Y.; Cai, L.; Chen, Y. A Raisin Foreign Object Target Detection Method Based on Improved YOLOv8. *Appl. Sci.* 2024, 14, 7295. [Google Scholar] [CrossRef]
14. Huang, L.Q.; Jiang, L.W.; Gao, X.F. An Improved YOLOv3-Based Real-Time Video Helmet Wearing Detection Algorithm. *Modern Computer* 2020, (30), 32–38+43.
15. Glorot, X.; Bengio, Y. Understanding the Difficulty of Training Deep Feedforward Neural Networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, Sardinia, Italy, 13–15 May 2010*; JMLR Workshop and Conference Proceedings: Cambridge, MA, USA, 2010; pp. 249–256.
16. Chen, Z.Q.; Zhang, Y.Q. An Improved Deep Sort Target Tracking Algorithm Based on YOLOv4. *Journal of Guilin University of Electronic Technology* 2021, 41(02), 140–145.

17. Jia, X.; Zhao, C.; Zhou, J.; Wang, Q.; Liang, X.; He, X.; Huang, W.; Zhang, C. Online detection of citrus surface defects using improved YOLOv7 modeling. *Trans. Chin. Soc. Agric. Eng.* 2023, 39, 142–151. [Google Scholar]
18. Zhang, M.H. Research on High-Robustness Multi-Pedestrian Tracking Algorithm Based on DeepSort Framework. Master's Thesis, Harbin Institute of Technology, Harbin, China, 2020.
19. Fang, W.; Ding, L.; Luo, H.; Love, P.E.D. Falls from heights: A computer vision-based approach for safety harness detection. *Autom. Constr.* 2018, 91, 53–61. [Google Scholar] [CrossRef]
20. Ding, L.; Fang, W.; Luo, H.; Love, P.E.D.; Zhong, B.; Ouyang, X. A deep hybrid learning model to detect unsafe behavior: Integrating convolution neural networks and long short-term memory. *Autom. Constr.* 2018, 86, 118–124. [Google Scholar] [CrossRef]
21. Chang, J.; Zhang, G.; Chen, W.; Yuan, D.; Wang, Y. Gas station unsafe behavior detection based on YOLO-V3 algorithm. *China Saf. Sci. J.* 2023, 33, 31–37. [Google Scholar]
22. Han, J.; Liu, Y.; Li, Z.; Liu, Y.; Zhan, B. Safety Helmet Detection Based on YOLOv5 Driven by Super-Resolution Reconstruction. *Sensors* 2023, 23, 1822. [Google Scholar] [CrossRef] [PubMed]
23. Ji, Z.; Zhou, Y.; Zhang, Y.; Guo, X.; Shi, K. Industrial site unsafe behavior detection based on improved YOLOv5. *China Saf. Sci. J.* 2024, 34, 38–43. [Google Scholar]
24. Li, J.; Hu, Y.; Huang, X.; Li, H. Small Target-Oriented Multi-Space Hierarchical Helmet Detection. *Comput. Eng. Appl.* 2024, 60, 230–237. [Google Scholar] [CrossRef]
25. Li, S.; Jiang, H.; Xiong, L.; Huang, C.; Jiang, Y.; Deng, X.; Wang, K.; Zhang, P. Hydraulic construction personnel safety helmet wearing detection algorithm based on STA-YOLOv5. *J. Chongqing Univ.* 2023, 46, 142–152. [Google Scholar]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.