Falling Objects/Debris Detection System Using Surveillance Camera at Construction Site

Songyuan Zheng, Meta Soy, Jinwoong Lee, Kyubyung Kang, Ph.D., Jiansong Zhang, Ph.D., and Emad Elwakil, Ph.D.

School of Construction Management Technology, Purdue University, USA
zheng776@purdue.edu, soym@purdue.edu, lee4067@purdue.edu, kyukang@purdue.edu, zhan3062@purdue.edu, eelwakil@purdue.edu

Abstract–

Falling objects and debris at construction sites pose significant safety hazards, often going unnoticed during operations. Identifying these incidents in post-construction reviews can enable safety managers to develop preventive action plans for future projects. This study uses existing surveillance cameras equipped with computer vision technology to detect, track, and count any fallen objects or debris. The proposed method achieved 100% accuracy across all metrics in Dataset 1, which was captured using a fixed camera. In contrast, Dataset 2, simulated with camera shake to mimic real-world conditions, showed decreased precision for smaller objects, with overall precision dropping to 95% and an F1 score of 98%. The system could be used to support the lesson-learned plan for future site safety project meetings, which could further minimize the potential objects/debris falling in construction projects.

Keywords –

Falling object; Falling debris; Safety; Computer vision.

1 Introduction

The construction site is a complex environment that could cause potential dangers to people working in the area. Especially with vertical structures such as buildings, falling debris could seriously harm the construction workers or other personnel. According to the International Safety Equipment Association (ISEA), a worker in the United State gets stuck by a falling object about every 10 minutes [1]. Those falling-debris or objects at construction sites could cause safety hazards resulting from potential injuries to casualties. To prevent those cases, the occupational safety and health administration (OSHA)'s guideline requires employers to equip workers with falling object protection gears, install necessary falling object protection systems, and guide on how to secure objects from falling [2]. Despite the effort, there are still accidents from falling objects happening.

With advancement in technology and availability of surveillance cameras in general, there are many studies trying to minimize construction site accidents by various means. For example, Yang et al. combined BIM with computer vision technology to detect the potential danger of falling when working near floor openings [3]. For the case of Personal Protection Equipment (PPE) detection, Zou et al. tested a light weight (You Only Look One, YOLOv5) model for helmet wearing detection when attached to the surveillance system [4]. In general, most research focuses on the prevention of accidents through developing a real time warning system [5], detecting the absence of PPEs [6], collision warning [7], monitoring falling of the workers [8] and so on.

Those studies mainly emphasize the planning and real-time case of the construction site safety. However, during the post-construction review, the feedback from all safety activities and data is critical for the safety manager to improve for the next project. Especially for the undetected or unreported accidents such as unnoticed falling debris/object was not mainly in the scopes of existing research, but that information is crucial in supporting the future safety action plan.

To address the gap, this study developed a falling objects/debris detection system using the OpenCV open-source tools, to detect, track and count the falling objects.

Two datasets are tested to validate the system. Dataset 1 is collected from a stable surveillance camera, while Dataset 2 is collected from a shaking surveillance camera to imitate the real construction site. The results show that for the Dataset 1, the precision, recall and F1 scores achieved 100%. However, for the Dataset 2, the system remained 100% recall, with a small drop in the precision (95%) and F1 score (98%). The results justify the system's effectiveness and its potential to support construction site safety strategies.

2 Literature Review

The study focuses on computer vision technology coupled with camera input to develop a system to count falling objects from the video data while the information is used for lessons learned in the post-construction review. Therefore, it is essential to investigate the technologies which were used for construction safety with computer vision applications. Based on the study done by Fang et al. (2020) [9] and Lee et al. (2023) [10], a variety of technologies have been developed for improving the safety in construction sites. In addition, the main enabling technology of computer vision is deep learning which is extensively used in many aspects including object detection, semantic segmentation and so on.

Deep learning is a multi-layer neural network architecture and in the case of computer vision, Convolutional Neural Networks (CNNs) allow computers to interpret and understand visual data. Fang et al. (2018) [6] used R-CNN architecture to input video from CCTV at construction site to detect workers who are not wearing safety hardhat. The proposed system obtains 95.7% precision and 94.9% recall respectively. Another popular object detection deep learning architecture is You Only Look One (YOLO). YOLO with various custom-made integrated layers is used in different safety conditions at construction sites. Zou et al. (2024) [4] proposed a YOLOv5 based light weight algorithm to detect workers wearing hats. The result suggests that the integration of the YOLOv5, Ghostbottle, and DWConv surpass the YOLOv5 model with 30% less model size. Despite good results in both accuracy and recall, the use of deep learning architecture requires massive amounts of datasets to train on. Therefore, regarding the falling object, Tu et al. (2024), inspired by the Moving Object Detection Dataset (MOD), was generated the called Falling Object Detection around Buildings (FADE) and FADE-net were proposed and trained on FADE dataset which contained fallen items such as bottles, packaging boxes, books, shoes, and so on [11].

While there is not much research using computer vision to detect falling objects and tracking them without the use of deep learning and dataset, there are some studies used OpenCV library to detect movement of any moving objects from video input. OpenCV library is a python open-source library which allows computers to visualize the object, allow the image processing, and is used with different machine leaning application for image detection, objection's motion, and so on. Pandey et al. [12] utilize contours to detect and track the motion of any moving object using OpenCV. Contours refer to the imaginary points on the boundaries of the focused object, highlighting two stages: object detection and object tracking. Through these stages, the authors can detect real-time objects frame by frame, and by connecting successive frames, they check the object's location in each frame and correlate them.

According to the above literature, and in the case of no dataset, using the computer vision with OpenCV approach is suitable for falling object/debris detection.

3 Methodology

The study aims to detect falling objects and count their number from surveillance camera footage. Unnoticed/unreported falling object/debris are recorded during the whole construction duration. The proposed algorithm enables the safety manager to easily review and count unreported falling object/debris after the construction by simply running algorithm with the video input. The information includes the falling object location based on the camera's field of view, number of objects, object types, and others can be used for post-construction safety analysis that will be key lessons learned for future projects.

The proposed algorithm employs the OpenCV package and consists of three steps, as depicted in Figure 1: Background Elimination, Object Detection, and Object Tracking.

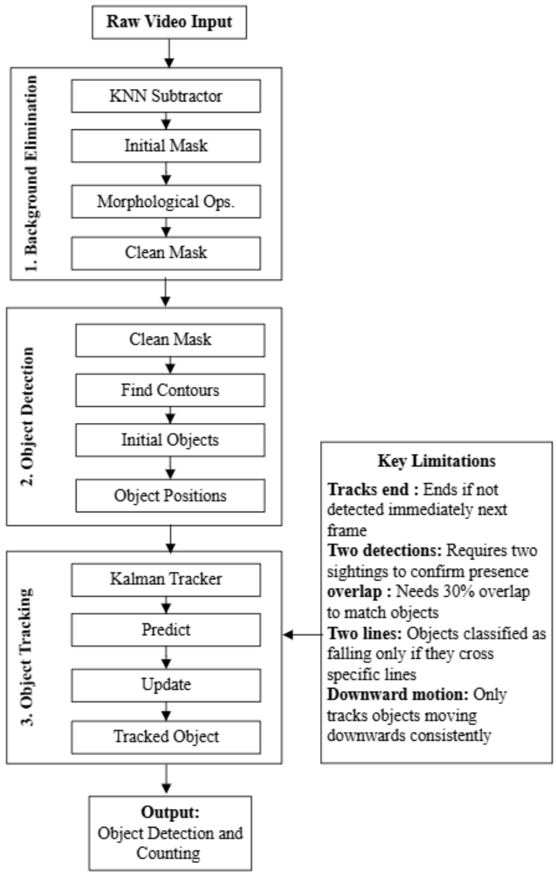

Figure 1. Proposed system architecture.

3.1 Background Elimination

This module aims to enhance the foreground scene for better analysis. The process begins with a background subtraction algorithm that extract moving foreground elements from a static background. This is followed by morphological operations—specifically, an opening to reduce noise and enhance object features within the mask by removing small, isolated elements. External contours of objects are then extracted from the refined mask to identify potential objects. Contours smaller than a predefined area threshold will be discarded to focus on significant objects. Bounding boxes are computed to save the detected objects in the remaining contours. The process isolates dynamic objects, enabling the following module: object detection and object tracking. Figure 2 shows how object's background is eliminated.

Figure 2. The effects of background elimination

3.2 Object Detection

In the object detection module, the cv2.findContours function is used to extract contours from the refined mask, specifically focusing on external contours to isolate significant object boundaries while ignoring internal variations. To avoid incorrect detections caused by camera vibration, the minimum area for contours is set to 10 pixels. This threshold ensures that smaller, potentially erroneous detections are disregarded, concentrating on more substantial and relevant objects. For the objects with the sizes beyond the minimum area, their bounding boxes (Figure 3) will be calculated and added to a list. This list will be saved and used in the next module.

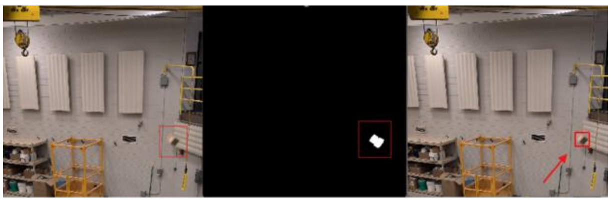

Figure 3. The effects of object detection.

3.3 Object Tracking

To track the falling objects, an object tracking module is developed based on the Kalman filter with an IoU threshold of 0.3.

The objects which show in the list, saved in the last step will be chosen to be tracked. In this process, the existing KalmanBoxTracker algorithm from OpenCV package will be used with the following parameters: track termination window will be one frame, meaning that the object tracking will be terminated if the module did not detect the falling objecting in the next frame. And the objects and its trajectory will be saved in another if they are detected in the continuous two steps.

Furthermore, to avoid some mistakes caused by the shaking camera, a two-line detection, where the objects could cross two lines will be detected as a falling object, is added to this module to ensure the detected objects with downward direction. the module also evaluates the angle of movement by analyzing the trajectory of the bounding box centers over ten frames. Only objects that consistently show a downward direction are detected as falling, ensuring precise detection and tracking of objects under specific motion conditions. An object is counted as falling only (shown in Figure 4) if it satisfies both the line-crossing and consistent downward trajectory criteria. These conditions are designed to accurately identify and track falling objects, avoiding the impact of a vibrating camera (or wind and other external forces).

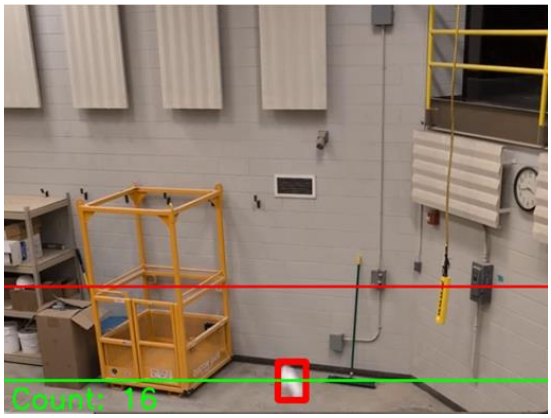

Figure 4. Detection and counting performance.

3.4 Evaluation Metrics

During the experiments, it was observed that some background elements were incorrectly identified as falling objects. In this context, evaluating the system's performance solely based on the ratio of detected objects to true falling objects would be misleading. Thus, to accurately assess the experimental results, Precision, Recall (also known as sensitivity), and F1 score are selected. The formulas for these metrics are provided in equations (1), (2), and (3). Correctly identified falling objects are True Positives (TP), while incorrect detections are False Positives (FP). Additionally, any falling objects that were not detected are considered False

Negatives (FN).

$$Precision = \frac{TP}{TP+FP} \quad (1)$$

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

$$F1 = 2 * \left(\frac{Precision*Recall}{Precision+Recall}\right) \quad (3)$$

4 Implementation and Results

The proposed system was evaluated using a dataset collected from the Moss Lab at Purdue University (Figure 5). The dataset consists of two subsets, totaling 20 videos. One subset (Dataset 1) was captured using a stable camera mounted on a tripod, providing controlled conditions. The other subset (Dataset 2) simulated camera vibration to mimic real-world construction site conditions. Objects were intentionally dropped from a height of 15 feet (as shown in figure 6) without controlling their speed to mimic the unplanned and accidental dropping of items on construction sites, recorded with a Google Pixel 9 Pro XL smartphone at 12MP resolution and 60 frames per second, were analyzed for detection and tracking accuracy. Each video featured five distinct objects, as shown in Figure 7.

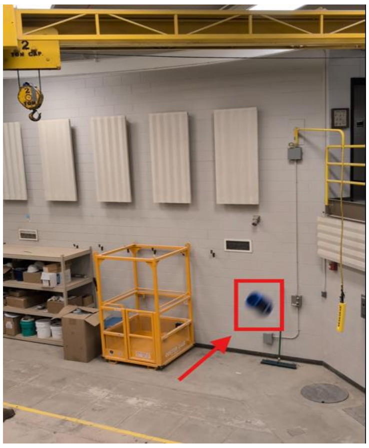

Figure 5. Example test of dropping object in Moss Lab.

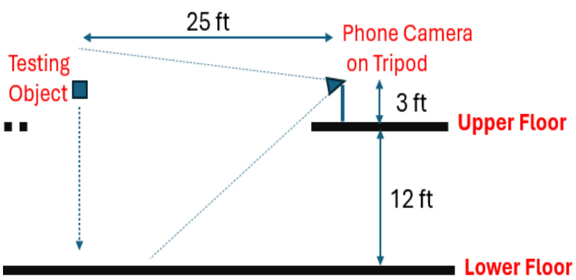

Figure 6. Testing setup layout (not to scale).

Figure 7. Object used for falling tests.

Under stable conditions (Dataset 1), the system achieved 100% precision, recall, and F1 scores for all objects as illustrated in Table 1. This demonstrates its high reliability in controlled environments, where object boundaries and movements are clearly distinguishable. The effectiveness of the system in Dataset 1 highlights its potential for detecting falling objects in situations where camera vibrations or environmental disturbances are minimal.

In dynamic conditions, Dataset 2, simulated through camera vibration, the system's performance slightly decreased for smaller objects. In Table 1, Precision for these objects dropped to 95% for the fourth object and 83% for the fifth object, while recall remained at 100% for all cases. This resulted in an overall precision of 95% and an F1 score of 98%. These results underscore the challenges posed by unstable camera conditions and the complexities of detecting smaller objects.

Figure 8 illustrates the challenges of false detections caused by background noise and camera vibration in Dataset 2. Subfigure 8a shows a static background element mistakenly identified as a falling object due to slight shifts in contour boundaries caused by vibrations. This misclassification demonstrates the system's sensitivity to motion artifacts, which can mimic object movement in the video. Subfigure 8b depicts another scenario where the system incorrectly identifies non-moving objects as falling, further highlighting the difficulties in isolating genuine object motion from dynamic background disturbances. These cases emphasize the importance of incorporating advanced filtering mechanisms to better handle background noise and reduce false positives.

Figure 9 provides an example of incorrect object trajectory classification under dynamic conditions. The figure shows an object misclassified as falling due to overlapping motion and inconsistent detection caused by vibrations. The yellow box and arrow highlight a non-falling object that was erroneously counted, revealing a limitation in the current trajectory analysis criteria. This issue is particularly prevalent in smaller objects, where fast movements and background inconsistencies can lead to misinterpretation. Enhancements such as refined trajectory angle analysis and improved stability algorithms could mitigate these errors.

Table. 1: Performance Evaluation on Dataset 1 and Dataset 2

Number of Objects / Size(m^3)		object 1 0.0106	object2 0.00295	object3 0.00378	object 4 0.0005	object5 0.000145	Total
Dataset 1	count (by our system)	10	10	10	10	10	50
	True Positive	10	10	10	10	10	50
	False Positive	0	0	0	0	0	0
	False Negative	0	0	0	0	0	0
	Precision	100%	100%	100%	100%	100%	100%
	Recall	100%	100%	100%	100%	100%	100%
	F1 score	100%	100%	100%	100%	100%	100%
Dataset 2	count (by our system)	20	20	20	21	24	105
	True Positive	20	20	20	20	20	100
	False Positive	0	0	0	1	4	5
	False Negative	0	0	0	0	0	0
	Precision	100%	100%	100%	95%	83%	95%
	Recall	100%	100%	100%	100%	100%	100%
	F1 score	100%	100%	100%	98%	91%	98%


(a) (b)

Figure 8. False detection due to camera movement.

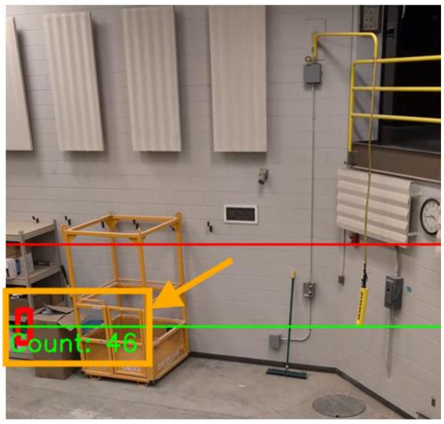

Figure 9. False detection sample.

The results demonstrate the system's robustness in detecting and tracking falling objects under stable conditions, achieving excellent performance metrics.

While the system remains effective under dynamic conditions, challenges such as false detections and reduced precision for smaller objects highlight areas for improvement. Addressing these limitations will be critical for deploying the system in real-world construction environments.

A complete and robust algorithm can be used by the safety manager for post construction analysis by simply detecting and counting unreported failing object/debris from input surveillance footage. The system can also be integrated in real-time with the live CCTV camera running all day. The detected failing object/debris can be considered as almost real time or daily feedback to safety managers which enable them to investigate or create any necessary action plan.

5 Limitation and Future Work

Despite achieving high performance on both testing datasets, the following limitations of the proposed systems are acknowledged: (1) the study experimented on five household objects, which might not accurately represent most objects at construction sites, which can vary in size and weight. (2) The experiment was set up indoors in the construction lab, which was susceptible only to the minimum wind, which could affect the motion of the falling object. (3) The camera set up and illumination conditions in the experiments, including the height and orientation of the camera as well as the laboratory lighting, could also vary.

The authors recommend future studies on conditions of different falling objects mainly taken or selected from construction sites. Direct testing in the construction site environment with existing CCTV cameras at different frame rates will be ideal. A real-time warning system could be developed based on the results of the proposed detection system. Specifically, the system would issue alerts via mobile devices whenever a falling object is detected by cameras positioned at various heights, ensuring that construction workers in different locations receive timely warnings. Additionally, the speed and number of falling objects recorded could be used to assess the risk level at the construction site, thereby enhancing safety measures and protocols.

6 Conclusion

Data about falling objects/debris during construction is crucial for lessons learned in the safety framework for construction projects. By leveraging the existing information from surveillance cameras and computer vision technology, the paper successfully proposed a system that can detect, track, and count falling objects/debris at construction sites. Based on the location of the CCTV camera field of view and the trajectory of the falling objects, the safety managers could analyze the cause of the falling case to improve the prevention/action plan further. According to the experimental results, the system shows good object tracking abilities with the counting 100 % precision, recall, and F1 for non-vibrating camera. In the camera vibration scenario, the system achieves 95% precision, 100% recall, and 98% F1 score, approximating real-world conditions more closely. Given the limitations of this study in the testing scenarios and testing environment, future study on different weight/types of falling objects, higher fame rate of the input video, and the use of different moving object detection algorithms are recommended.

7 References

[1] N. Randall. ISEA Release Updated Dropped Object Standard. On-line: https://safetyequipment.org/isea-releases-new-dropped-objects-standard/, Accessed: 12/22/2024.

[2] Occupational Safety and Health Administration, Fall Protection Systems and Falling Object Protection, OSHA Standard 1910.29, 2023. On-line: https://www.osha.gov/laws-regs/regulations/standardnumber/1910/1910.29. Accessed: Dec. 22, 2024.

[3] B. Yang, B. Zhang, Q. Zhang, Z. Wang, M. Dong, and T. Fang, Automatic detection of falling hazard from surveillance videos based on computer vision and building information modeling, Structure and Infrastructure Engineering, vol. 18, no. 7, pp. 1049-1063, 2022.

[4] X. Zou, Z. Chen, and Y. Zhou, A Lightweight Model Based on YOLOv5 for Helmet Wearing Detection, Proceedings of SPIE, 2024. doi: 10.1117/12.2627279

[5] Y. Jeon, D. Q. Tran, A. S. Kulinan, T. Kim, M. Park, and S. Park, Vision-based motion prediction for construction workers safety in real-time multi-camera system, Advanced Engineering Informatics, vol. 102898, 2024.

[6] Q. Fang, H. Li, X. Luo, L. Ding, H. Luo, T. M. Rose, and W. An, Detecting non-hardhat-use by a deep learning method from far-field surveillance videos, Automation in Construction, vol. 85, pp. 1–9, 2018.

[7] W. Gan, K. Gu, J. Geng, C. Qiu, R. Yang, H. Wang, and X. Hu, A Novel Three-Stage Collision-Risk Pre-Warning Model for Construction Vehicles and Workers, Buildings, vol. 14, no. 8, p. 2324, 2024.

[8] A. Danilenka, P. Sowiński, K. Rachwał, K. Bogacka, A. Dąbrowska, M. Kobus, K. Baszczyński, M. Okrasa, W. Olczak, P. Dymarski, I. Lacalle, M. Ganzha, and M. Paprzycki, Real-Time AI-Driven Fall Detection Method for Occupational Health and Safety, Electronics, vol. 12, no. 20, p. 4257, 2023.

[9] W. Fang, L. Ding, P. E. D. Love, H. Luo, H. Li, F. Peña-Mora, B. Zhong, and C. Zhou, Computer vision applications in construction safety assurance, Automation in Construction, 110: 103013, 2020.

[10] J. Lee and S. Lee, Construction Site Safety Management: A Computer Vision and Deep Learning Approach, Sensors, 23(2): 944, 2023.

[11] Z. Tu, Z. Gao, Z. Zhang, C. Zhou, J. Yuan, and B. Du, "FADE: A Dataset for Detecting Falling Objects around Buildings in Video," arXiv, 2024.

[12] H. Pandey, P. Choudhary, and A. Singh, Detect and Track the Motion of Any Moving Object Using OpenCV, in R. Agrawal, J. He, E. Shubhakar Pilli, and S. Kumar, Eds., Cyber Security in Intelligent Computing and Communications, Studies in Computational Intelligence, vol. 1007, Springer, Singapore, 2022.