Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control
Journal homepage: http://kinetik.umm.ac.id
ISSN: 2503-2267
Vol. 10, No. 4, November, Pp. 603-612

Development of a web-based information system for real-time fainting detection using YOLO in smart healthcare

Wiwit Agus Triyanto*[1], Nanik Susanti[1]
Department of Information System, Faculty of Engineering, Universitas Muria Kudus, Kudus, Indonesia[1]

Article Info

Keywords:
Web-based Information System, Fainting, YOLOv12, Computer Vision, Smart Healthcare

Article history:
Received: July 08, 2025
Accepted: September 12, 2025
Published: November 01, 2025

Cite:
W. A. Triyanto, "Development of a Web-based Information System for Real-time Fainting Detection Using YOLO in Smart Healthcare", *KINETIK*, vol. 10, no. 4, Nov. 2025.
https://doi.org/10.22219/kinetik.v10i4.2407

*Corresponding author.
Wiwit Agus Triyanto
E-mail Address:
at.wiwit@umk.ac.id

Abstract

Loss of consciousness (fainting) is a critical condition that requires prompt treatment, especially in the context of elderly health services and independent patient care. This research aims to develop a web-based information system that is able to detect fainting events in real-time using the You Only Look Once (YOLO) algorithm version 11, which is one of the latest approaches in deep learning-based object detection. The system is designed to monitor video from the surveillance camera directly, make visual inferences of the patient's posture, and provide automatic notifications if a loss of consciousness condition is detected. The dataset was obtained from the Roboflow platform and consists of 9,081 annotated images representing the fainting position. The YOLOv11 model was trained and tested using training data sharing, validation, and testing methods. The test results showed that the model achieved mAP, precision, recall and F1-score values of 98.70%, 98.00%, 97.30% and 97.65%, respectively. The developed information system is able to display the detection visually through the bounding box on the dashboard and record the time of the incident. With this performance, this system shows great potential in improving patient safety through intelligent monitoring and automated response in hospital, nursing home, and residential environments. This research also opens up opportunities for the development of more adaptive AI-based health monitoring systems and computer vision in the future.

1. Introduction

The rapid development of computer vision and artificial intelligence (AI) has had a major impact on various fields, with the healthcare sector being one of the most potential areas. In clinical settings, computer vision allows for automatic monitoring and analysis of patient activities, reducing reliance on manual observation and increasing response speed in emergency situations [1]. Computer vision technology is increasingly used in the healthcare sector [2] to detect critical events such as fainting or loss of consciousness—conditions that require immediate medical attention to prevent further complications. Various models of computer vision algorithms have been developed due to the growing interest in object identification, detection, and recognition [3]. The implementation of treatment in healthcare settings requires the ability to accurately track and identify objects, which is essential for directly monitoring patient activities, such as detecting fainting or falls.

The development of computer vision system integration in healthcare settings provides various benefits, including continuous monitoring, reduced reliance on manual observations, and faster response times in emergency situations that require immediate attention. The importance of real-time processing is exposed, especially in time-sensitive events, such as respiratory failure, where every second is critical in providing relief, increasing the likelihood of desired outcomes, and preventing potential hazards [4]. Remote monitoring for rural patients is carried out using telehealth technology, including services for the elderly who use wearable devices [5]. The application of machine learning and artificial intelligence in the visualization of patient health metrics has the potential to improve the support provided to healthcare practitioners.

The development of real-time detection systems for critical events, such as loss of consciousness or fainting, is gaining traction due to its potential to significantly improve safety and well-being across a wide range of domains [6]. In particular, computer vision techniques offer a non-invasive and efficient approach to continuously monitor individuals and detect early signs of such events [7]. Among various object detection algorithms, the *You Only Look Once* (YOLO) family has emerged as a popular choice for real-time applications due to its speed and accuracy [8]. The pursuit of replicating human vision on computers is essential to advance everyday technology and enable machines to understand and interpret visual data [9]. Object detection, a critical aspect of computer vision, enables the automatic identification and localization of objects in images or video frames, laying the groundwork for a wide range of applications, including autonomous vehicles, surveillance systems, and medical diagnostics [10]. This development is particularly relevant in dealing with time-sensitive events, such as cases of fainting in a nursing home or aged care facility, where every second counts. On the other hand, integration with web-based information systems provides new opportunities in the provision of real-time monitoring solutions. Web technologies such as WebRTC and WebSocket enable live video streaming, instant notification delivery, and responsive dashboards that can be accessed across devices. This integration ensures scalability, remote access, and operational efficiency in the implementation of smart healthcare systems.

Camera-based (non-wearable) systems are widely used because they are non-intrusive, although they still face challenges in the form of occlusion, lighting variations, and privacy issues. On the other hand, wearable and telehealth approaches offer high accuracy yet rely on patient compliance. The latest trend is leading to multi-modal systems that combine cameras with health sensors to improve clinical reliability. The proposed system utilizes the YOLO algorithm, which is renowned for its speed and accuracy in object detection. YOLO divides the image into grid cells, creating a bounding box around the recognized item. The system's ability to immediately identify an incident of fainting can substantially improve patient safety and well-being in a healthcare setting [11]. Real-time monitoring of patients' vital signs and algorithms can further automate and improve the detection of clinical malfunctions in the ward environment [12]. This approach reduces the likelihood of delayed interventions and improves the overall quality of care provided [13].

Although research in the field of fall detection has developed rapidly, the detection of fainting (syncope/fainting) is still relatively under-reported. In fact, fainting has a different dynamic than a regular fall, often occurring suddenly, and can pose a serious risk if not treated immediately. Delays in detecting these events can lead to delayed medical treatment and lower the chances of patient safety. In addition, most of the existing systems are not yet real-time [14] and are not designed to be web-native, resulting in limitations in video streaming and instant notifications. Another issue is the limitated dataset for fainting cases, which leads to low generalization performance of the model. On the operational side, many studies only report on model accuracy (e.g. mAP or F1-score), without considering practical metrics such as time-to-alert, or false alarm rate, which are very important in real-world applications. As such, there is an urgent need to develop a web-based, real-time fainting detection system that can operate quickly and accurately within a smart healthcare environment, thereby improving the safety and quality of patient care.

To answer these problems, this study proposes the development of a web-based information system for real-time detection of fainting by utilizing the YOLO algorithm. This algorithm was chosen for its ability to balance inference speed and detection accuracy, making it suitable for clinical application that demand immediate response. The system will utilize the optimized YOLOv11 architecture through quantization techniques and compression models to maintain computing efficiency without compromising detection performance [15]. The system is designed with a web-based architecture that allows for real-time video streaming integration using WebRTC, instant notification delivery via WebSocket, and responsive dashboards. This approach supports cross-device access, both by medical personnel and patients' families, while ensuring system scalability in a hospital or community-based healthcare setting.

In addition, the study also emphasizes the evaluation of time-to-alert and false alarm rate, in addition to traditional accuracy metrics. Thus, the solutions offered not only focus on the success of the detection model, but also ensure the readiness of real implementation in the clinical world. By combining YOLO's cutting-edge algorithms and integrated web architecture, the system is expected to improve patient safety through faster, more accurate, and more accessible fainting detection. The main contribution of this solution is to provide a real-time, automated, and accessible approach to improve patient safety, reduce reliance on manual observation, and accelerate medical intervention in emergency situations.

2. Research Method

Development of a web-based computer vision information system to detect loss of consciousness using YOLOv11 involves several key steps. This method starts with the acquisition of relevant datasets [16], which include video footage of individuals experiencing loss of consciousness as well as control situations. After obtaining the dataset, the video is pre-processed to improve image quality and consistency. The YOLOv11 model was then trained using a pre-processed dataset to identify indicators of loss of consciousness [17], such as sudden, unresponsive, or unusual movements.

The training procedure entails refinement of model parameters to improve detection accuracy and reduce false positives [18]. The YOLOv11 model can produce a higher level of precision and performance. After training, the YOLOv11 model is integrated into a web-based system that allows real-time video stream analysis. The YOLOv11 model is implemented as a core detection engine in a web-based system, using its architecture to process video frames efficiently and accurately. The system is designed to receive video data from a variety of sources, including webcams, surveillance cameras, and pre-recorded video files. The implementation process involves several software components and hardware configuration.

The system is built using scalable and reliable web technologies, such as Flask, to handle high volumes of concurrent users and data streams. The front-end interface is built with HTML, CSS, and JavaScript, offering users an easy-to-use platform to access detection results and manage system settings. The system architecture combines load balancing and automatic scaling to ensure optimal performance and availability, especially during peak usage times.

The system includes real-time video streaming, object detection visualization, and alert management. The real-time video streaming capability allows users to view the video feed directly from the connected camera, while the object detection visualization overlays the detection results to the video stream, showing a bounding box around the recognized object. The alert management feature allows users to configure notifications for specific events, such as loss of consciousness detection, ensuring that the right response is triggered immediately. The research method is shown in Figure 1 below.

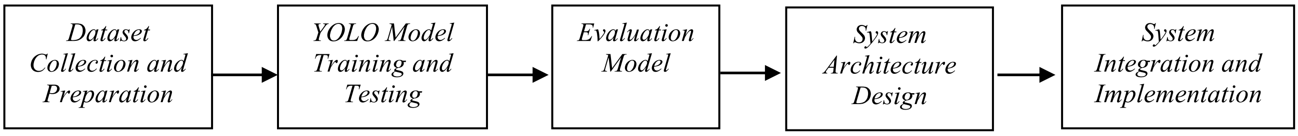

Figure 1. Research Methods

2.1 Dataset Collection and Preparation

The quality and relevance of the datasets used for model training and evaluation are critical [19] in the development of a real-time fainting detection system using the YOLO method. The dataset serves as the foundation in building reliable object detection models to recognize fainting events with high accuracy.

The dataset was collected using a high-quality camera and a public dataset called "Fainting Detection", which is available on the Roboflow Universe platform, resulting in a total of 9,081 annotated images showing people in various positions of fainting or losing consciousness. This dataset is licensed under CC BY 4.0, which allows its use for academic and research purposes by including relevant features. Each image is annotated by marking the label "fainted" if it is detected to faint. An example of the fainting dataset used in this study can be seen in Figure 2.

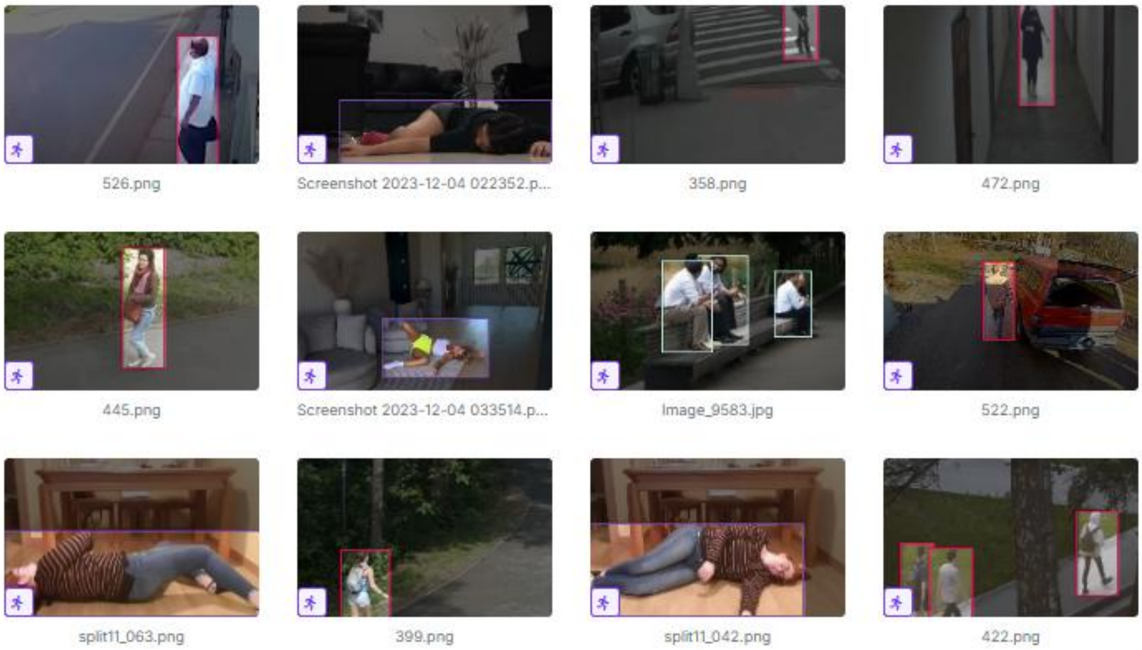

Figure 2. Example of Fainting Detection Dataset

Before the model training was carried out, the collected datasets were preprocessed and augmented, which can be seen in Table 1.

Table 1. Parameters for Preprocessing and Augmentation

Preprocessing	Auto-Orient: Applied
	Resize: Fit within 640x640
Increases	Outputs per training example: 8
	Flip: Horizontal, Vertical
	Crop: 0% Minimum Zoom, 25% Maximum Zoom
	Rotation: Between -10° and +10°
	Grayscale: Apply to 20% of images

	Saturation: Between -20% and +20%
	Brightness: Between -20% and +20%
	Blur: Up to 1px

2.2 YOLO Model Training and Testing

The proposed system utilizes the YOLO algorithm, which is renowned for its speed and accuracy in object detection [20]. YOLO divides the image into grid cells, creating a bounding box around the recognized item [21]. The system's ability to immediately identify an incident of fainting can substantially improve patient safety and well-being in a healthcare setting. Real-time monitoring [22] of patient vital signs and algorithms can further automate and improve the detection of clinical malfunctions in the ward environment. This approach reduces the likelihood of delayed interventions and improves the overall quality of care provided.

The datasets that have been annotated, preprocessed, and augmented are then divided into three main subsets: training data (70%), validation data (20%), and testing data (10%). This division is carried out to maintain a balance between the model training process, parameter adjustment, and final evaluation objectively. This study uses the YOLOv11 model, the latest variant of the YOLO family, which offers a lightweight architecture, fast inference performance, and full support for a wide range of dataset formats. The initial parameters used include: 16 Batch Size, 100 Epochs, and 0.001 Learning Rate. Datasets that have been processed in YOLO format are used as inputs. The training process includes: Forward propagation, Loss calculation, Backpropagation and Checkpoint saving where the model is stored at a specific epoch with the best performance (based on mAP).

2.3 Evaluation Model

The evaluation of the YOLO model uses metrics to measure various aspects of its performance, including efficiency and accuracy [10]. Mean Average Precision (mAP) serves as the basic metric that provides a broad measure of object detection accuracy across different categories of objects [23]. The mAP calculates the average precision for each class and then calculates the average value for each class, providing a thorough assessment of the model's ability to correctly detect and localize objects [24]. Two basic metrics, Precision and Recall, provide an idea of the difference between a correctly identified object and an actual proportion of an object detected. Recall measures the accuracy of positive predictions, which show the proportion of objects correctly identified among all objects predicted by the model. Precision, on the other hand, measures the completeness of positive predictions, which indicate the proportion of actual objects correctly identified by the model [25]. The F1 score, which is the harmonic mean of precision and recall, provides a balanced measure of accuracy. This metric is particularly useful when dealing with imbalanced datasets, where one class has significantly more instances than the other. The evaluation metrics of mAP, Precision, Recall, and F1-score are shown in Equations 1-4.

$$mAP@_{\alpha} = \frac{1}{n}\sum_{i=1}^{n} AP_i \ for\ n\ classes. \tag{1}$$

$$precision = \frac{TP}{TP + FP} \tag{2}$$

$$recall = \frac{TP}{TP + FN} \tag{3}$$

$$F1 = 2 * \frac{1}{\frac{1}{precision} + \frac{1}{recall}} \tag{4}$$

2.4 System Architecture Design

This information system is designed as a web-based platform that integrates: Fainting detection model using YOLO, Browser-based user interface, Backend server for data processing & inference, and Database for storage of detection results. The system design is illustrated through a use case diagram, which can be seen in Figure 3.

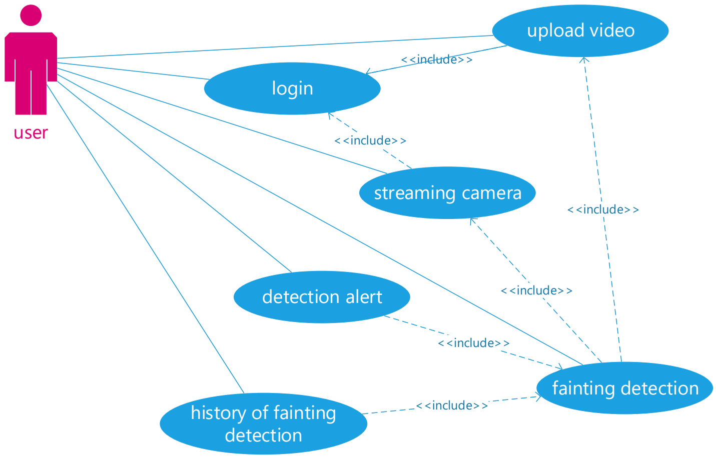

Figure 3. Use Case Web-based Fainting Detection Information System

From the use case diagram, it can be seen that the main component in this system is the user actor who can do several actions such as logging in, uploading video or streaming from a camera, viewing fainting detection, viewing fainting detection history, and receiving fainting detection notifications. Meanwhile, the functional system needs analysis can be seen in Table 2.

Table 2. Functional System Needs Analysis

Module	Technology	Main Functions
Fainting Detection	YOLOv11, WebRTC, WebSocket	Detect fainting objects from video streams or file uploads
Backend API	Flask	Provide detection services and connections to the frontend
Database	MySQL	Storing detection results and user data
Frontend Web	HTML5, CSS3, JavaScript	Display the monitoring dashboard

2.5 System Integration and Implementation

The system consists of two main components that are integrated, namely YOLO Detection Engine and a web-based monitoring system. YOLO Detection Engine serves as a detection model that uses a trained YOLOv11 model, running on the server backend to process video input or camera pounds in real time. Meanwhile, the web-based monitoring system functions as a web-based user interface that allows operators to see live streaming video from the camera, video upload, detection results (status and bounding box), and history of fainting events.

The integration of the YOLO model into the server backend is carried out by first exporting the trained YOLOv11 model into .pt format. The backend is developed using a Python framework, Flask, which serves as an API provider for the model inference process. Video streaming originating from the camera is processed, where each frame received is sent to the YOLO model for object detection. The result of this process is bounding box coordinates and object class labels which are then passed to the system logic module for further handling of data storage.

Integration with web-based information systems is carried out by building a backend using Python (Flask) that connects to the MySQL relational database to store detection data and user information. On the frontend side, the user interface is developed using HTML5, CSS3, and JavaScript. The detected data, including the time of the incident, location, and imagery captured by the system, is dynamically displayed on the monitoring dashboard to provide real-time information to users.

3. Results and Discussion

The results of the implementation show that web-based information systems were successfully developed by utilizing a client–server architecture, which allows for the flexibility of access through different devices without the need for additional installations. This approach is in line with research [5], which emphasizes that web-based systems with client–server architecture support are able to improve scalability and efficiency in intelligent healthcare. The successful use of this architecture ensures that the system can be used both in a hospital environment and in the context of remote patient monitoring. In addition, the integration of the YOLO (You Only Look Once) algorithm to detect fainting events in real-time shows significant performance. These results support the findings [8], stating that YOLO is one of the superior object detection paradigms due to its speed and precision, making it ideal for time-sensitive applications such as emergency healthcare. With this integration, the system can immediately recognize fainting events and provide early warning, potentially reducing delays in medical interventions.

From the interface side, the system is equipped with an intuitive monitoring dashboard, featuring live video streaming features and automatic notifications when a fainting event is detected. This feature strengthens the results of the research [4], which emphasizes the importance of real-time monitoring mechanisms and instant notifications to increase the speed of response to medical emergencies. Thus, the developed interface serves not only as an information center, but also as an effective means of early warning. Furthermore, the system supports two main input sources, namely real-time cameras and video file uploads, providing flexibility in its use. A similar approach has also been put forward by [12], which suggests that a combination of input from live sources and video archives can improve the reliability of clinical detection systems. With this multi-input support, the system can adapt to a wide range of situations, both for direct monitoring of patients in healthcare facilities and for analysis of previous medical records.

3.1. Detection Accuracy and Performance

After training using the YOLOv11 model on a dataset of 9,081 annotated images, the system achieved a fairly good detection performance. Based on the results of the evaluation on the test data, the model achieved an mAP of 98.70%, with a precision of 98.0%, a recall of 97.30%, and an F1-score of 97.65%. Meanwhile, the YOLOv8 model has lower results with an mAP of 99.10%, a precision of 96.90%, a recall of 97.50% and an F1-score of 97.20%. These results show that the YOLOv12 model is capable of detecting fainting events with high accuracy and low false detection rate. The detailed results are presented in Table 3.

Table 3. Results of the Evaluation of the YOLOv8 and YOLOv12 Models

	mAP	Precision	Recall	F1 Score
YOLOv8	99,10%	96,90%	97,50%	97,20%
YOLOv11	98,70%	98,00%	97,30%	97,65%

This achievement is in line with research [2], emphasizing that the development of computer vision algorithms, including YOLO, has further improved the performance of object identification, detection, and recognition in a real-world context. Thus, the successful implementation of YOLOv11 in this study confirms that the method is feasible for use in smart health applications that require a high level of precision. In terms of performance, the system is capable of running at real-time speeds, reaching an average of 30 frames per second (FPS) on standard hardware. This performance demonstrates good processing efficiency, supporting the implementation of patient monitoring without significant lags. These findings are consistent with research [15], which states that the YOLO family has advantages in speed, computing efficiency, and relatively lightweight architecture, making it suitable for real-time applications. The speed is a crucial factor in detecting medical emergencies such as fainting, where every second is critical to patient safety.

3.2 Web-based system integration

The integration of web-based systems in this study provides ease of access and flexibility for users. The system can run directly through a standard browser without the need for additional installation, supporting a wider range of implementations across various devices, including computers, tablets, and smartphones. This approach is consistent with the findings [5], emphasizing that the utilization of web technology in telehealth allows for remote patient monitoring with more cost-efficient and wider service coverage. Being web-based, the system is also easier to update and maintain compared to conventional desktop applications. Key features integrated into the system include instant alert notifications as well as a real-time monitoring dashboard that displays detection results from the YOLO algorithm. Instant notifications displayed through the web interface allow medical personnel to respond more quickly to emergencies. This is in line with research [4], which emphasizes the importance of real-time-based early warning systems to reduce delays in medical responses to time-sensitive events, such as fainting or respiratory failure.

In addition, this system is also equipped with automatic storage of event logs within the database. Each detection is stored with supporting information, such as time, date, input source, and links to video recording details. This approach is in line with the findings [12], which suggests that systematic management of health data through web-based database integration can help improve the accuracy of diagnosis and the quality of medical interventions. With a combination of real-time monitoring, instant notifications, and data storage features, the integration of web-based systems in this study provides a comprehensive and adaptive solution to support smart healthcare.

Figure 4 shows the system interface in the file upload menu section which is designed to make it easier for users to perform video analysis automatically. Through this menu, users can select and upload video files stored on their devices to be processed by the YOLO-based fainting detection system. The interface is presented in a simple and intuitive way, equipped with a "Choose File" button to select a video as well as an "Upload" button to send files to the server. Once the file is successfully uploaded, the system will run the analysis process and display the detection results on the monitoring dashboard.

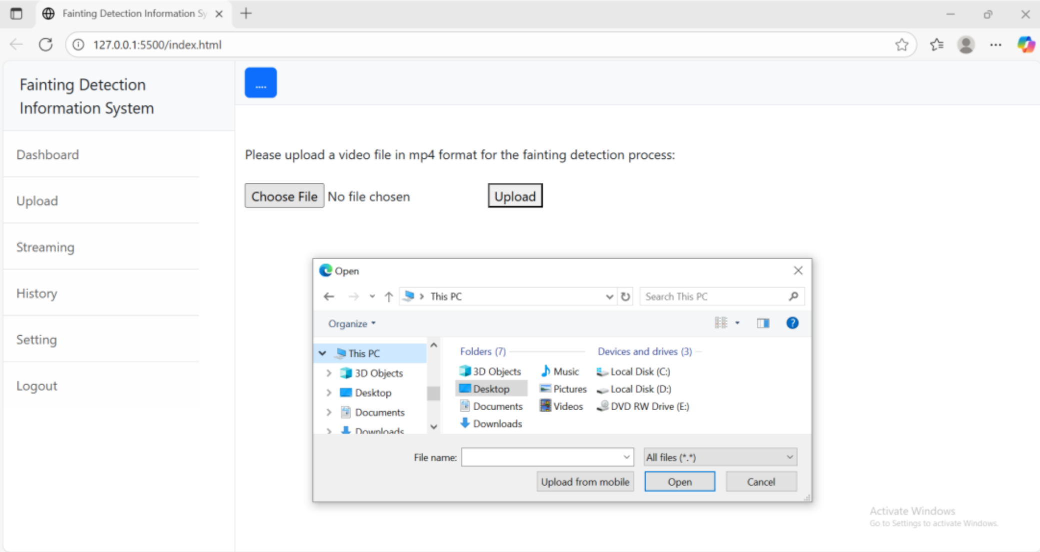

Figure 4. File Upload Menu Display

Figure 5 shows the system interface used to monitor the patient's condition directly through a real-time camera. In this menu, users can access live streaming from a connected camera, and the system automatically runs the YOLO algorithm to detect fainting events instantly. If an indication of fainting is detected, the system will provide a warning notification, allowing medical personnel to intervene immediately. This feature is essential in supporting continuous patient monitoring without having to perform manual observations continuously. In addition, the use of web-based streaming technology enables cross-device access through standard browsers, allowing it to be used on both computers and mobile devices.

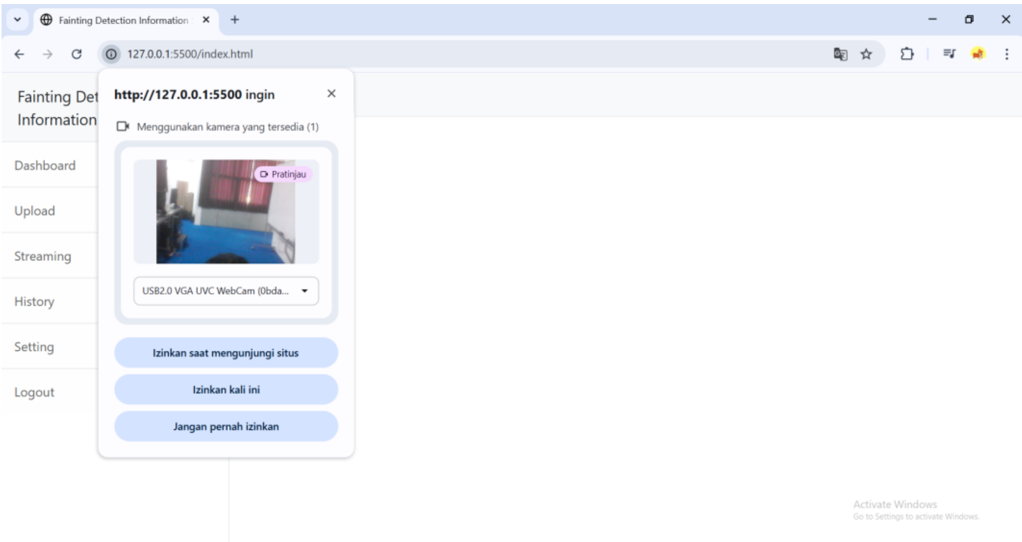

Figure 5. Streaming Camera Menu Display

Figure 6 shows the results of the implementation of the YOLO algorithm in detecting fainting events in real-time. In this display, the system automatically provides a marker in the form of a purple bounding box that surrounds the detected object, accompanied by a Fainted label to clarify the patient's condition. This visualization is proof that the system is able to accurately identify critical events and immediately display them on the monitoring dashboard. With clear markings, medical personnel can more quickly recognize emergency situations and take necessary intervention steps. The integration of bounding boxes and detection labels is also in line with previous research, where visual representations are considered effective in increasing the speed of decision-making in the field of smart healthcare.

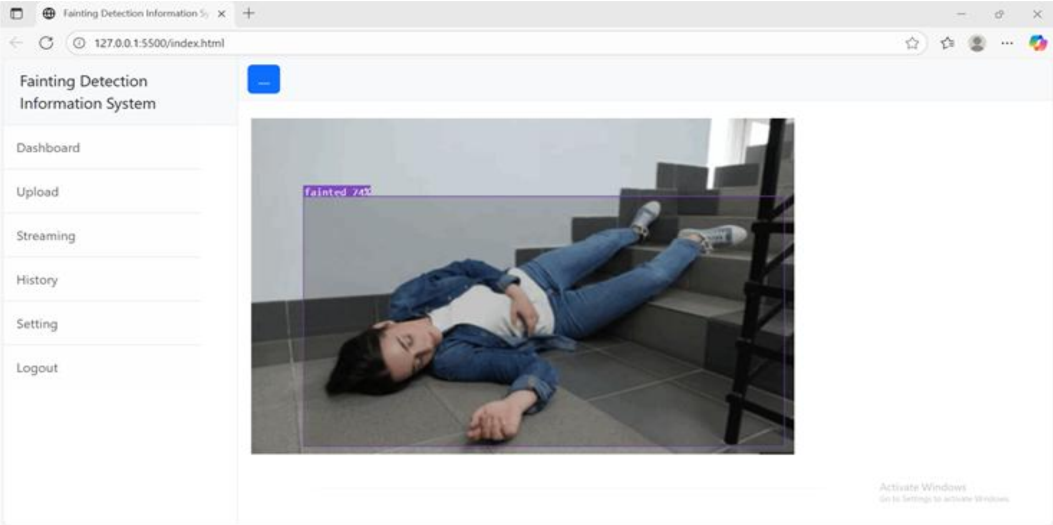

Figure 6. Fainting Detection Display

Figure 7 shows the history of the fainting detection that have been recorded by the system. In this menu, each entry contains detailed information in the form of time (hours), date, detection status (true or false), and data sources used, both from real-time cameras and video file uploads. This history feature allows medical personnel or system managers to retrace incidents that have occurred, making it easier to analyze and evaluate.

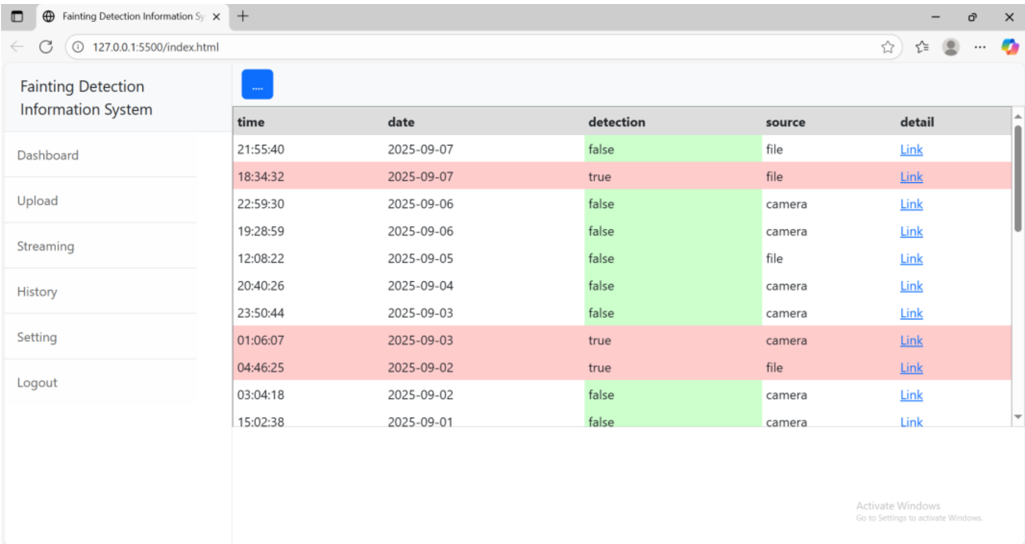

Figure 7. Fainting History Detection Display

Figure 8 shows the automatic notification feature that the system generates when a fainting event is detected. This notification appears in real-time on the web interface, with a clear warning message that says "Detected Fainting". The presence of notifications serves as an early warning for medical personnel or supervisors, enabling them to take quick action immediately. The notification design is made simple but firm, making it easy to recognize even when the user is accessing other features on the dashboard.

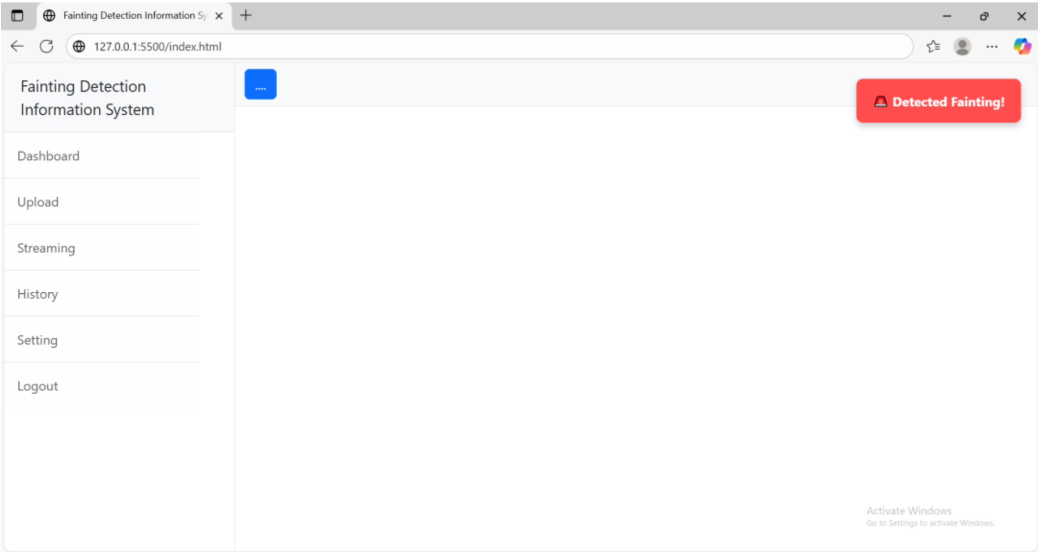

Figure 8. Notification Detection Fainting Display

3.3 Advantages and Limitations

The analysis shows that the developed system has a number of advantages over manual observation methods. With YOLO-based automatic detection, the system is able to reduce delays in responding to fainting events, as notifications are given in real-time to medical personnel or supervisors. These findings are in line with research [13], which emphasizes that automation in clinical detection can significantly improve the quality of patient care while reducing the likelihood of delayed interventions. Additionally, the implementation of serverless architecture in this system provides advantages in terms of scalability and cost efficiency. This approach allows the system to operate with dynamic resources as needed, without the reliance on traditional server infrastructures that tend to be expensive and complex. This is in line with the findings [9], explaining that cloud-based technology innovations and serverless architecture provide a great opportunity to improve the flexibility and performance of smart health systems.

Another advantage that stands out is the responsive web that allows the system to be accessed across platforms through a variety of devices with similar user experiences. This feature ensures that the system remains usable in environments with both hardware and software limitations. [5] It also emphasized the importance of cross-device accessibility in telehealth to expand the reach of health services, particularly for people in remote areas. Thus, these advantages confirm that the developed system is not only effective, but also adaptive to the needs of users in various healthcare contexts.

Although the system shows promising performance, there are some limitations that need to be noted. One of the main challenges is the detection performance in extreme lighting conditions or when there are obstructive objects in front of the camera. In situations like this, the accuracy of the system decreases because the model has difficulty recognizing the patient's posture clearly. These findings emphasize that limited lighting and visual quality are critical factors that still limit the performance of computer vision-based systems in health monitoring. In addition, the system also has the potential to produce false positives, especially when the patient performs movements that resemble fainting, such as sitting up suddenly or looking down quickly. This can lead to unnecessary alarms and increase the workload of medical personnel. This limitation is in line with research [3], which notes that even though modern object detection algorithms are becoming more precise, complex differences in human movement can still trigger detection errors. Therefore, it is necessary to develop a more contextual model, for example by combining posture analysis sequentially to differentiate between fainting and normal activity.

System performance is also still greatly influenced by the capacity of the hardware, both in terms of servers and cameras used. If run on low-spec devices, the frame rate may drop and reduce the effectiveness of real-time monitoring. This is in line with research [10], emphasizing that computing efficiency and hardware optimization are important aspects to ensure stable performance in vision-based healthcare applications. Thus, although this system is effective, optimization strategies are still needed at both the algorithm and infrastructure levels to minimize current limitations.

4. Conclusion

This research has resulted in a web-based information system that uses the YOLOv11 object detection algorithm in real-time. It enables healthcare workers to quickly and accurately recognize the occurrence of loss of consciousness by combining computer vision technology with an interactive monitoring interface. With mAP, precison, recall and F1-score values of 98.70%, 98.00%, 97.30% and 97.65%, respectively, this system can be applied in a smart healthcare environment. Live video monitoring, visual tagging with bounding box, and automatic recording to the database can be performed effectively by this system.

User test results show that the dashboard interface is easy to understand and easy to use for everyday tasks. Although the results are promising, there are some limitations. Some of these are lower accuracy in low-light conditions and the possibility of false positives when the user is in an unusual body position. Advanced development can include support for more than one camera, the use of temporal analysis for event validation, and the adaptation of the system for more complex environments. Overall, the system contributes to intelligent health monitoring solutions, especially for hospitals, nursing facilities, and home-based patient care. By incorporating YOLO into a web-based platform, artificial intelligence can improve responsiveness, security, and automation in healthcare.

References

[1] R. Shinta, J. Jasril, M. Irsyad, F. Yanto, and S. Sanjaya, 'Classification of Leaf Disease Images of Rice Plants Using CNN with VGG-19 Architecture', *J. Science and Inform.*, vol. 9, no. 1, pp. 37–45, 2023. http://doi.org/10.22216/jsi.v9i1.2175

[2] J. Liu, J. Guo, and S. Zhang, 'YOLOv11-HRS: An Improved Model for Strawberry Ripeness Detection', *Agronomy*, vol. 15, no. 5, p. 1026, 2025. http://doi.org/10.3390/agronomy15051026

[3] Q. Aini, N. Lutfiani, H. Kusumah, and M. S. Zahran, 'Object Detection and Recognition with Machine Learning Models: Yolo Model', *CESS (Journal Comput. Eng. Syst. Sci.*, vol. 6, no. 2, p. 192, 2021. http://doi.org/10.24114/cess.v6i2.25840

[4] K. Vijayaprabakaran, K. Sathiyamurthy, and M. Ponniamma, 'Video-Based Human Activity Recognition for Elderly Using Convolutional Neural Network', *Int. J. Secur. Priv. Pervasive Comput.*, vol. 12, no. 1, pp. 36–48, 2020. http://doi.org/10.4018/ijsppc.2020010104

[5] T. Shaik *et al.*, 'Remote patient monitoring using artificial intelligence: Current state, applications, and challenges', *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, vol. 13, no. 2, 2023. http://doi.org/10.1002/widm.1485

[6] S. Shen, Z. Wu, and P. Zhang, 'Research on target detection method of distracted driving behavior based on improved YOLOv8', *arXiv (Cornell Univ.*, 2024. http://doi.org/10.48550/arxiv.2407.01864

[7] J. H. Yousif, 'Fuzzy logic approach for detecting drivers' drowsiness based on image processing and video streaming', 2021. http://doi.org/10.36227/techrxiv.16777927

[8] N. Adiuku, N. P. Avdelidis, G. Tang, and A. Plastropoulos, 'Advancements in Learning-Based Navigation Systems for Robotic Applications in MRO Hangar: Review', *Sensors*, vol. 24, no. 5, p. 1377, 2024. http://doi.org/10.3390/s24051377

[9] R. Huang, J. Pedoeem, and C. Chen, 'YOLO-LITE: A Real-Time Object Detection Algorithm Optimized for Non-GPU Computers', *2021 IEEE Int. Conf. Big Data (Big Data)*Pp. 2503–2510, 2018. http://doi.org/10.1109/bigdata.2018.8621865

[10] N. Jegham, C. Y. Koh, M. Abdelatti, and A. Hendawi, 'Evaluating the Evolution of YOLO (You Only Look Once) Models: A Comprehensive Benchmark Study of YOLO11 and Its Predecessors', *arXiv (Cornell Univ.*, 2024. http://doi.org/10.48550/arxiv.2411.00201

[11] Y.-H. Li, Y. Li, M.-Y. Wei, and G. Li, 'Innovation and challenges of artificial intelligence technology in personalized healthcare', *Scientific Reports*, vol. 14, no. 1. Nature Portfolio, 2024. http://doi.org/10.1038/s41598-024-70073-7

[12] A. Brankovic *et al.*, 'Explainable machine learning for real-time deterioration alert prediction to guide pre-emptive treatment', *Sci. Rep.*, vol. 12, no. 1, 2022. http://doi.org/10.1038/s41598-022-15877-1

[13] B. J. Erickson, P. Korfiatis, Z. Akkus, and T. L. Kline, 'Machine Learning for Medical Imaging', *Radiographics*, vol. 37, no. 2. Radiological Society of North America, pp. 505–515, 2017. http://doi.org/10.1148/rg.2017160130

[14] K. Kumar and K. M. B. A. Safwan, 'Accelerating Object Detection with YOLOv4 for Real-Time Applications', *arXiv (Cornell Univ.*, 2024. http://doi.org/10.48550/arxiv.2410.16320

[15] A. Rasheed and M. Zarkoosh, 'YOLOv11 Optimization for Efficient Resource Utilization', *arXiv (Cornell Univ.*, 2024. http://doi.org/10.48550/arxiv.2412.14790

[16] M. Mao and M. Hong, 'YOLO Object Detection for Real-Time Fabric Defect Inspection in the Textile Industry: A Review of YOLOv1 to YOLOv11', *Sensors*, vol. 25, no. 7. Multidisciplinary Digital Publishing Institute, p. 2270, 2025. http://doi.org/10.3390/s25072270

[17] T. E. Lockhart *et al.*, 'Prediction of fall risk among community-dwelling older adults using a wearable system', *Sci. Rep.*, vol. 11, no. 1, 2021. http://doi.org/10.1038/s41598-021-00458-5

[18] W. Gong, 'Lightweight Object Detection: A Study Based on YOLOv7 Integrated with ShuffleNetv2 and Vision Transformer', *arXiv (Cornell Univ.*, 2024. http://doi.org/10.48550/arxiv.2403.01736

[19] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, 'You only look once: Unified, real-time object detection', in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788. http://dx.doi.org/10.1109/CVPR.2016.91

[20] J. Terven and D. Cordova-Esparza, 'A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS', *arXiv (Cornell University)*. Cornell University, 2023,. http://doi.org/10.48550/arxiv.2304.00501

[21] W. A. Triyanto, K. Adi, and J. E. Suseno, 'Indoor Location Mapping of Lameness Chickens with Multi Cameras and Perspective Transform Using Convolutional Neural Networks', *Math. Model. Eng. Probl.*, vol. 11, no. 2, pp. 539–548, 2024. http://doi.org/10.18280/mmep.110227

[22] A. T. Y. Chen, J. Fan, M. Biglari-Abhari, and K. I. K. Wang, 'A computationally efficient pipeline for camera-based indoor person tracking', *Int. Conf. Image Vis. Comput. New Zeal.*, vol. 2017-Decem, pp. 1–6, 2018. http://doi.org/10.1109/IVCNZ.2017.8402479

[23] M. Yaseen, 'What is YOLOv8: An In-Depth Exploration of the Internal Features of the Next-Generation Object Detector', *arXiv (Cornell Univ.*, 2024. http://doi.org/10.48550/arxiv.2408.15857

[24] G. A. Pereira, 'Fall Detection for Industrial Setups Using YOLOv8 Variants', *arXiv (Cornell Univ.*, 2024. http://doi.org/10.48550/arxiv.2408.04605

[25] A. S. Geetha, 'Comparing YOLOv5 Variants for Vehicle Detection: A Performance Analysis', *arXiv (Cornell Univ.*, 2024. http://doi.org/10.48550/arxiv.2408.12550