ICIC Express Letters
Volume **18**, Number **3**, March **2024**

 ISSN 1881-803X
pp. **209–218**

VISION-BASED FALL DETECTION USING CONVOLUTION NEURAL NETWORK AND LONG SHORT-TERM MEMORY

Patipat Sitpasert[1], Atchara Namburi[2] and Worawat Sa-Ngiamvibool[1,*]

[1]Faculty of Engineering
Mahasakham University
41/20 Kham Riang, Kantharawichai, Mahasarakham 44150, Thailand
65010353004@msu.ac.th; *Corresponding author: worawat.s@msu.ac.th

[2]Faculty of Science and Engineering
Kasetsart University
59 Moo 1, Chiang Krua, Muang, Sakon Nakhon 47000, Thailand
csnarn@ku.ac.th

Received April 2023; accepted July 2023

Abstract. *This paper presents a method of vision-based model for detecting and classifying human falls in video sequences. We used BlazePose to detect and extract 33 body landmarks of a human body; then, we selected 4 points to represent the upper body. Then, we draw a straight line "r" to calculate the angle of the upper body, linear velocity, and angular velocity to help determine if the person detected has fallen. These data are similar to the data obtained from gyroscope and accelerometer sensors. We then use the capabilities of CNN and LSTM to construct a model for fall detection. In addition, we used DeepSORT to track people in the video and identify who fell. We conducted experiments on three datasets, and our model achieved a high accuracy rate of 96.66%, recall of 89.95%, the precision of 96.72% and F1-score of 93.08%.*
Keywords: Fall detection, Deep learning, DeepSORT, CNN, LSTM

1. **Introduction.** Thailand is becoming an ageing society, where many families have elderly members who live alone or are left alone when their children go to work. Falls in older people are one of the most dangerous accidents that threaten their health in daily life compared to other age groups, as seniors have a higher risk of falls due to physical weakness, frailty, and poor balance. In those aged 65 years and older, the risk of falls is 28%-35%, increasing to 32%-42% in those aged 70. According to the statistics from the Ministry of Public Health of Thailand in 2018 [1], approximately 30% of seniors fall at least once a year, and 1.67% of those injured from falling die annually. Globally, falls are the second leading cause of accidental or unintentional injury deaths worldwide and are responsible for about 684,000 deaths per year [2]. Early detection of falls can help reduce the severity of injuries and improve the affected individuals' outcomes. Falls can cause severe injuries for older people and lead to death without immediate assistance. However, falls can be less severe if medical treatment arrives promptly. Given the seriousness of this issue, it is crucial to develop a system that can detect falls and send immediate notifications to reduce the duration of harm. The sooner help arrives, the lower the morbidity and mortality. Thus, the authors propose developing a detection and notification system that is highly accurate.

In the last decade, numerous surveillance systems have been developed to detect and alert against falls. These systems aim to reduce injuries in senior citizens and prevent severe injuries from falls. Fall detection systems can generally be categorized into two main

DOI: 10.24507/icicel.18.03.209

types: hardware-reliant-based and video-based. Hardware-based systems use sensors attached to the person being monitored, such as accelerometers, gyroscopes, or pressure sensors. These sensors can detect changes in the person's movement and orientation, indicating a fall event. Hardware-based systems can provide high accuracy; however, false alarms are also high and can be intrusive, uncomfortable, and require regular maintenance and battery replacement. Video-based fall detection systems, on the other hand, use cameras and computer vision algorithms to analyze video footage and identify specific patterns that indicate a fall event. These systems can be less intrusive and more user-friendly, as they do not require physical contact with the monitored person. Additionally, video-based systems can provide additional contextual information about the fall event, such as the location and direction of the fall, which can be valuable for healthcare professionals and caregivers.

This work was inspired by [3] that used 2D upper body representation and SVM (Support Vector Machine) to classify falls. Despite its high accuracy, the authors mentioned limitations in detecting falls that occur forwards or backward or while lying down, where the detection results could have been better. Therefore, this paper presents a vision-based model for detecting and classifying human falls without sensors. We used BlazePose to detect and extract 33 body landmarks of a human body; then, we selected 4 points and drew a straight line "r" (see Figure 2(b)) to represent the 3D upper body. Next, we calculate the angle of the upper body, linear and angular velocity. These data are similar to the data obtained from gyroscope and accelerometer sensors. We then sent the data to a CNN-LSTM (Convolutional Neural Networks – Long Short-Term Memory) model to classify human falls. The model will have high accuracy, similar to sensor-based systems. Not using sensors can reduce the cost of developing models and also helps reduce problems with discomfort, forgetting to wear maintenance and battery charging.

We outline the structure of the remaining sections of the paper. Section 2 reviews the existing literature on state-of-the-art fall detection. Section 3 introduces our ideas and methods for vision-based fall detection. Section 4 presents the results of the work and discusses its findings. Finally, Section 5 summarizes the main points and discusses future research directions.

2. **Related Works.**

There has been a growing interest in developing automated fall detection systems using different technologies in recent years. This literature review will go over the other fall detection systems and their effectiveness.

2.1. **Wearable sensors.**

Wearable sensors offer continuous monitoring and alert caregivers or emergency responders in the event of a fall. Several studies have explored the use of wearable sensors for fall detection, e.g., [4] researches that detects the persons posture and activities, and the following review summarizes some of the key findings. Studies [5, 6] have demonstrated the combining of accelerometers and machine learning and they obyained a high sensitivity and specificity in fall detection. Various sensor modalities show potential for fall detection. For example, gyroscopes can detect angular velocity variations indicative of falls [7], while barometers can sense changes in air pressure during falls [8]. Optimal sensor placement on the body has also been studied, with trunk or waist placement suggested as more effective, although limb placement can still achieve high accuracy [6]. However, false alarms can pose a concern in fall detection as sudden movements or other activities may be mistakenly identified as falls [5]. Additionally, frequent charging requirements, discomfort associated with wearable devices, and potential side effects [9] may render them unsuitable for older individuals. Some older individuals may experience self-consciousness or embarrassment when wearing sensors, leading to reduced compliance and discontinuation of use [10]. Moreover, cost considerations make wearable sensors relatively expensive, particularly for low-income individuals.

2.2. **Vision-based.** Vision-based fall detection has been an active research area in recent years due to its potential to provide low-cost and non-intrusive fall detection solutions [11]. Traditional vision-based methods aim to detect real-time falls by analyzing video frames. These approaches utilize computer vision techniques to extract significant features, such as silhouettes [12], bounding boxes [13], deflection angles [3] or aspect ratio [14], from the frames. Researchers have employed various techniques to identify falls, including shape matching [15] or head tracking [16]. Conventional video-based approaches often require subject extraction, which can be affected by image noise. However, the emergence of deep neural networks has improved detection performance in vision-based fall detection. Researchers have utilized Convolutional Neural Networks (CNNs), combined CNNs with Long Short-Term Memory (LSTM) networks [17], and incorporated visual attention mechanisms to extract spatiotemporal features and effectively detect falls. These advancements have shown promise in enhancing the accuracy and reliability of fall detection systems. A study by [17] achieved an impressive accuracy of 99.73% on a benchmark dataset using this method. However, lighting conditions [18] and occlusions [19] can affect the performance of vision-based fall detection systems, leading to false positives or false negatives. To address this issue, researchers have explored the use of additional sensors, such as infrared [20] and depth cameras [21]. Another essential aspect to consider in vision-based fall detection systems is privacy. The use of cameras to monitor individuals can raise privacy and consent concerns. A study by [22] proposed a privacy-preserving fall detection system using a Generative Adversarial Network (GAN) to generate synthetic images for fall detection, which reduces the privacy concerns associated with using authentic images. While the effectiveness of vision-based systems can be influenced by factors such as lighting conditions, camera placement, and algorithm performance, there are situations where sensor-based systems might be preferred, such as in low-light environments or when prioritizing privacy. Nonetheless, vision-based fall detection methods offer numerous advantages. They are non-intrusive since individuals do not need to wear additional sensors. These methods analyze video frames in real time, enabling prompt fall detection and rapid response. Vision-based systems have broad coverage, making them suitable for monitoring multiple individuals simultaneously. Using existing camera infrastructure, they can also serve various purposes, including video surveillance and activity recognition. Additionally, these systems can consider environmental factors contributing to falls, providing insights for improved safety. Vision-based fall detection is cost-effective as it utilizes existing cameras, eliminating the need for extra hardware or wearables. Moreover, vision-based systems are particularly well-suited for older adults who often experience forgetfulness and are less prone to injuries and irritations.

3. **Proposed Method.** CNN-LSTM fall detection is a technology that uses deep learning techniques to detect falls in real time. The system combines the Convolutional Neural Network (CNN) and the Long Short-Term Memory (LSTM) network to analyze skeleton data from BlazePose [23], the real-time pose detection system and detect patterns that indicate a fall. The CNN-LSTM fall detection system typically uses data such as angle, linear velocity and angular velocity of the upper body segments of a person. These values will be computed using four key landmarks extracted from skeleton data using the deep learning library BlazePose. This data is then processed through the CNN to extract features and patterns indicative of falls, such as sudden changes in velocity or orientation. The LSTM analyzes the time-series data and detects patterns that may indicate a fall. A typical output of the system comprises three distinct classes that respectively indicate the pre-fall, falling, and post-fall states. If a fall is detected, the system can trigger an alarm or alert a caregiver or emergency services. CNN-LSTM fall detection technology can improve the safety and independence of older adults and individuals with mobility

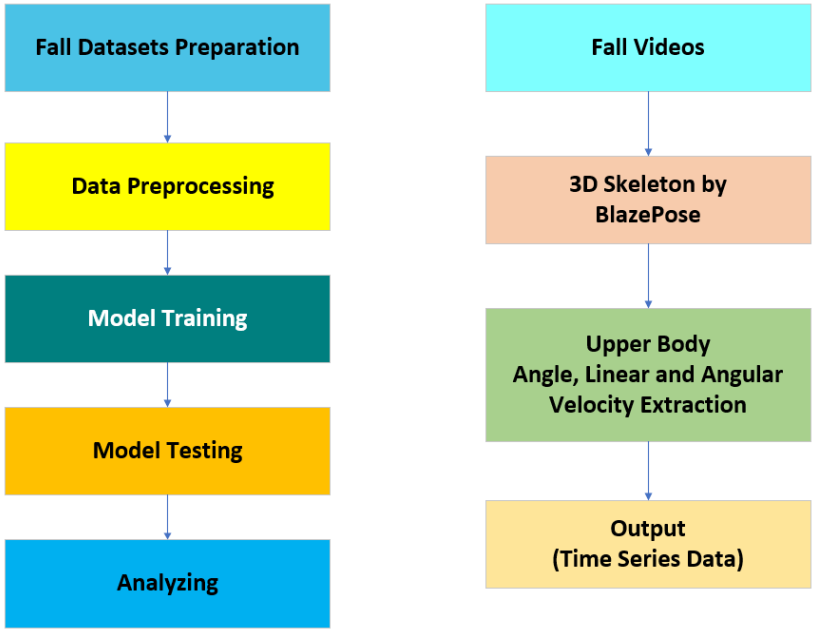


FIGURE 1. (a) Overview of our proposed system; (b) the data preparation step

impairments. By detecting falls quickly and accurately, the system can enable prompt medical attention and reduce the risk of serious injury or complications.

The research process has been divided into five steps as follows and can be presented in Figure 1(a).

1) **Video data preparation (Datasets):** In this stage, we provide video data by utilizing three commonly used datasets in state-of-the-art research. There are 228 colour videos from three different datasets used for our work, 160 videos for training, and 68 videos for testing.

a) ImVia Fall Detection Dataset (ImVia) [24]: The dataset which consists of 191 videos recording both normal activities and falls in different locations, such as homes, cafes, offices, and classrooms.

b) UR Fall Detection Dataset (URFD) [25]: The dataset consists of 70 videos, wherein 30 videos are focused explicitly on falls, while the remaining 40 capture everyday activities. For our research, we exclusively utilized the 30 videos that depict falls, as this is the area of interest for our study. Falls are recorded using 2 Kinect cameras, while normal activities are recorded using only one camera.

c) FallAllD [26]: A Comprehensive Dataset of Human Falls and Activities of Daily Living, which consists of 7 videos, this dataset covers falls and daily activities.

2) **Data preprocessing:** To preprocess the data, we use BlazePose to extract the primary 3D skeleton. We compute the significant values of the upper body angle, linear velocity, and angular velocity of a person's upper body segments. These values will create time series data that can be used in the CNN-LSTM model.

3) **Model training:** We use Training Datasets (70%) to develop an accurate predictive model. The model is then evaluated using Testing Datasets (30%) to gauge its performance and accuracy.

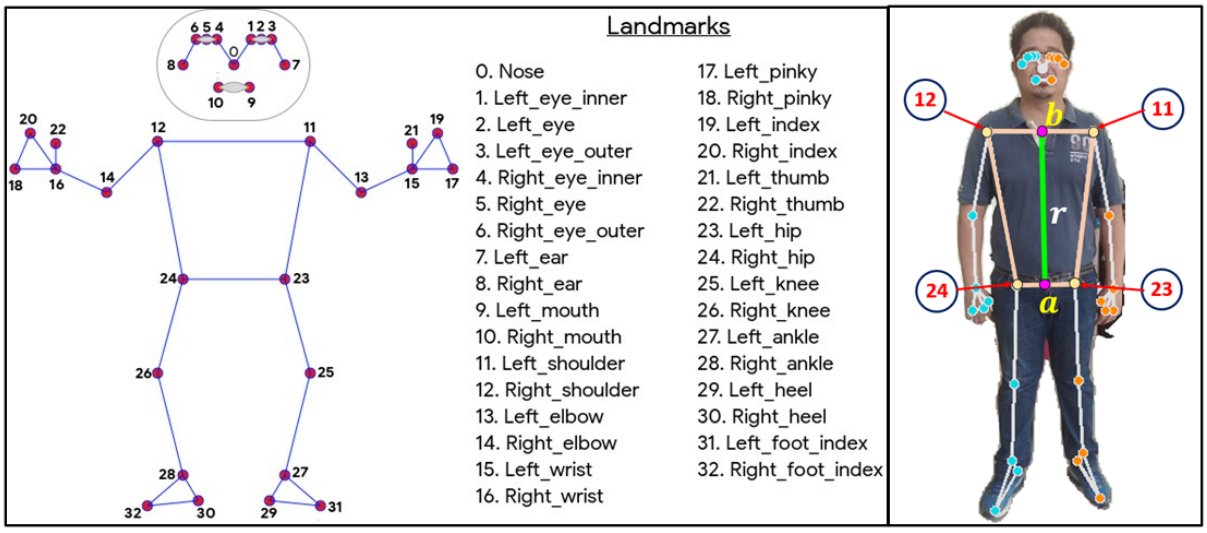
4) **Model testing:** In this step, we test the model created by applying it to Test Videos to evaluate the performance.

5) **Analyzing:** In this step, we analyze the test results and evaluate the model's performance to determine its accuracy and effectiveness in predicting outcomes. Test result analysis includes accuracy, precision, recall and F1-score.

3.1. **Data preprocessing.** To standardize the video data used for our experimental analysis, we have adjusted it to ensure a consistent frame rate of 25 frames per second. Subsequently, we extracted the video data into a series of images, each with a standardized width of 320 pixels. Finally, we label to determine if the person is falling. The falls will be categorized into three stages: pre-fall (consisting of standing, sitting, walking, or other), falling, and post-fall (lying down). Finally, the videos will be divided into two categories: Train Videos (70%) for model training and Test Videos (30%) for further model testing.

The process involves two primary steps to get the data ready for the research, shown in Figure 1(b). The first step consists in using BlazePose to extract 3D skeletal data. In contrast, using four crucial landmarks, the second step involves computing the Angle, Linear Velocity, and Angular Velocity of a person's upper body segments. These values will create time series data that can be used in the CNN-LSTM model.

3.2. **3D Skeleton.** Individuals will be detected, and their 3D skeletal data will be extracted using BlazePose, an open-source platform from Google for creating cross-platform machine learning solutions that can be customized for live streaming, highlighting its speed. The resulting data will be 33 sets of 3-dimensional (x, y, z) data points corresponding to 33 positions on the body, as shown in Figure 2(a). Figure 3 shows samples of a 33-landmark obtained from the BlazePose model.



(a) (b)

FIGURE 2. (a) A 33-landmark obtained from the BlazePose model; (b) a 4-landmark and a straight line r representing the upper body

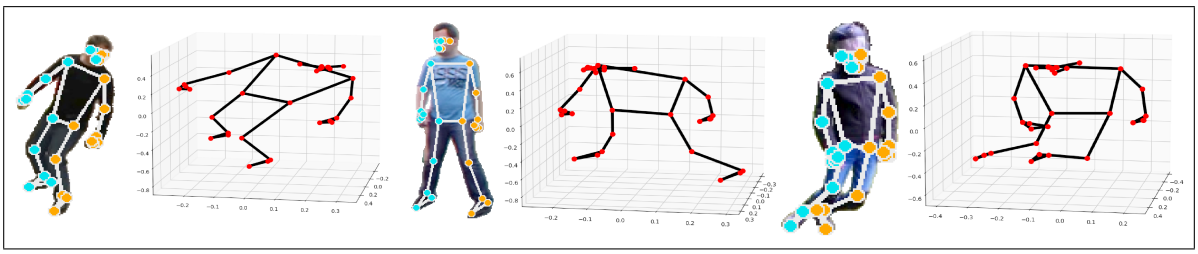


FIGURE 3. Samples of a 33-landmark obtained from the BlazePose model

3.3. **Features calculation.** We picked 4 points from the 33 in the 3D Skeleton data. These points are point 12 (right shoulder), point 11 (left shoulder), point 24 (right hip), and point 23 (left hip), which can be seen in Figure 2(b). We used these 4 points to create

a straight line r, which runs from point a (halfway between points 23 and 24) to point b (halfway between points 11 and 12). This line r is used to represent a person's upper body.

Equations (1) and (2) are applicable for computing the body angle (ϕ, θ, ψ) of the upper part of the human body. This angle is determined about the x, y, and z axes, as illustrated in Figure 4(a). The linear velocity $(\dot{x}_b, \dot{y}_b, \dot{z}_b)$ of point b (neck) and the angular velocity $\left(\dot{\phi}, \dot{\theta}, \dot{\psi}\right)$ can be determined from Equations (3) and (4), respectively.

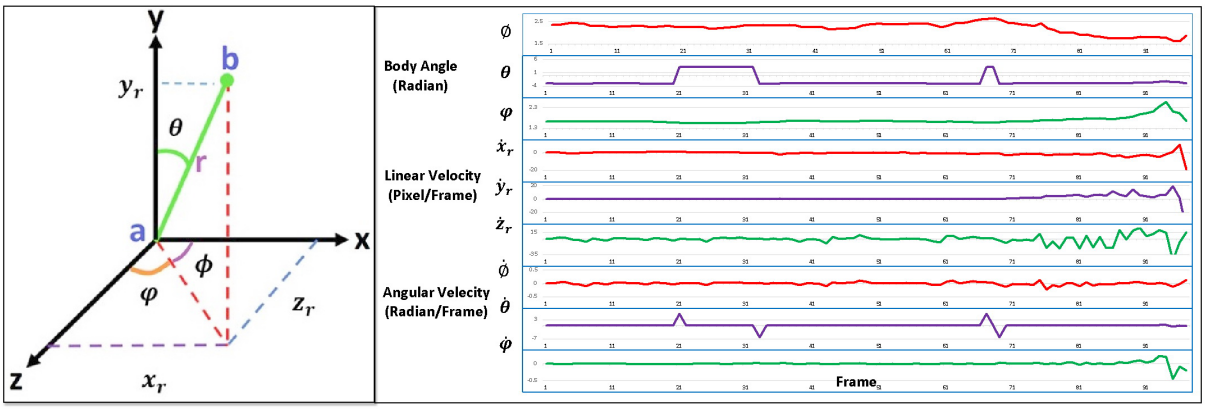
$$\bar{r} = (x_b - x_a, y_b - y_a, z_b - z_a) \tag{1}$$

$$A = (\phi, \theta, \psi) = \left(atan\left(\frac{z_r}{x_r}\right), acos\left(\frac{y_r}{|\bar{r}|}\right), atan\left(\frac{x_r}{z_r}\right)\right) \tag{2}$$

$$V = (\dot{x}_b, \dot{y}_b, \dot{z}_b) = (x_{bk} - x_{bk-1}, y_{bk} - y_{bk-1}, z_{bk} - z_{bk-1}) \tag{3}$$

$$\dot{A} = \left(\dot{\phi}, \dot{\theta}, \dot{\psi}\right) = (\phi_k - \phi_{k-1}, \theta_k - \theta_{k-1}, \psi_k - \psi_{k-1}) \tag{4}$$

The information gathered from each video will be saved in a time-series format, consisting of 9 rows. This information will include the angles of the upper body parts, the linear velocities and the angular velocities, as displayed in Figure 4(b). Once all the information is gathered for each frame, it will be divided into 24 frames extended (1 second) windows with an overlap of every 12 frames (0.5 seconds). Then, the data will be combined with the information from all the other videos, randomized, and split into two parts: 70% for training the model and 30% for testing it.

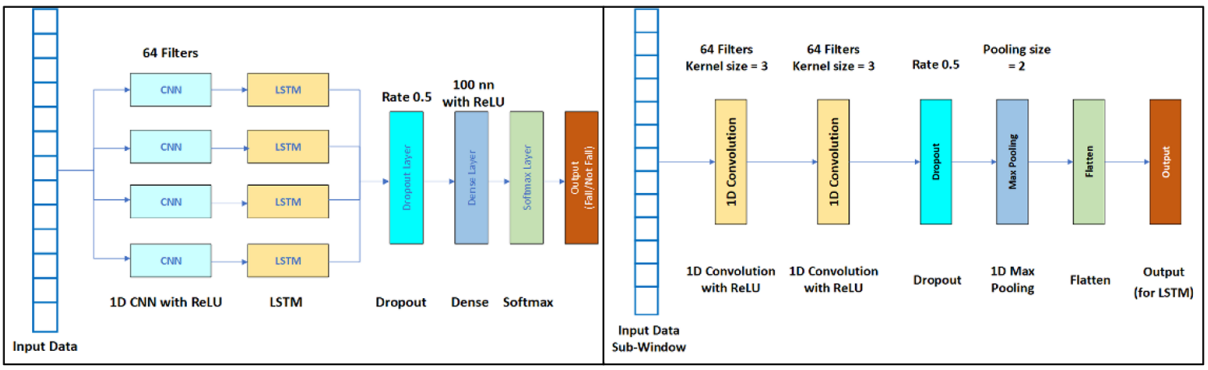


(a) (b)

FIGURE 4. (a) The straight line r and angles of the axes ϕ, θ, and ψ; (b) data in a time series format used to train a model using CNN-LSTM

3.4. **CNN-LSTM model.** The data obtained from pre-processing the training videos are split into two sets. 70% of the data is used for training the model. In contrast, the remaining 30% is used for evaluating the model's performance. To identify human activities such as falling, we employed a CNN-LSTM model, which extracts features using a CNN and sequences the meaning of those features over time using an LSTM recurrent neural network. The research aims to detect whether a person falls in the video. Figure 5(a) illustrates the model's structure.

The input data is a 1D array with nine rows and 24 frames (1 window) divided into four equal parts and fed into the CNN model, which has a structure shown in Figure 5(b). The CNN model has two layers of 1D convolution with 64 filters and a kernel size of 3, followed by a Dropout with a rate of 0.5 to reduce overfitting. After that, 1D Max Pooling with a size of 2 and Flatten are applied. The CNN model output is then fed into the LSTM model to sequence the meaning of features over time and recognize various



(a) (b)

FIGURE 5. (a) The CNN-LSTM model for fall detection; (b) Convolution Neural Network (CNN) model

human activities, including falling. Then, three layers of Dropout with a rate of 0.5, Dense, and Softmax, follow the LSTM model, respectively.

3.5. **Model testing.** The model trained in the previous step will be tested on a 30% subset of the prepared video data. In this study, we are interested in tracking individuals to determine their status and whether they have fallen. Furthermore, the data fed to the CNN-LSTM model is in a continuous time series format for each object. Therefore, it is necessary to continuously track each object to reduce errors resulting from the swapping of data during fall detection. Consequently, we employed DeepSORT as the tracking algorithm for this research, and YOLOv5 was used as the object detection algorithm. Human movement in the video is assumed to follow a Hidden Markov Model (HMM).

3.5.1. *HMM.* A hidden Markov model [27] will be used to define the movement of people in the video, with objects defined as bounding boxes obtained from object detectors. The state $x_t = \left[c_x, c_y, \gamma, h, c'_x, c'_y, \gamma', h'\right]$ is defined, where c_x and c_y are the centre points along the x and y axes, respectively, γ is the aspect ratio of the bounding box, h is the height, and c'_x, c'_y, γ', h' are the velocities of each variable, respectively.

3.5.2. *YOLOv5.* To detect objects (people) and obtain bounding box results for object tracking using the DeepSORT algorithm, we chose to use the YOLOv5 object detection model trained on the COCO dataset [28], which can detect objects in 80 categories with high speed and accuracy. According to [29], when tested on the Tesla P100 dataset, YOLOv5 processed images at approximately 0.007 seconds per image or 140 Frames Per Second (FPS), making YOLOv5 suitable for real-time object detection.

3.5.3. *DeepSORT.* DeepSORT [30] is an object-tracking algorithm based on the SORT framework, which is designed for use in video sequences. It improves on SORT (Simple Online and Realtime Tracking) by incorporating deep neural networks for object detection and feature extraction, as well as a re-identification module that enables tracking of temporarily occluded or out-of-view objects. The framework also employs the Kalman filter [31] for predicting object states and the Hungarian algorithm for object association. We used DeepSORT to track objects in our videos and assign unique IDs to individuals. This allowed us to determine whether each person had fallen and categorize falls as pre-fall, fall, or post-fall.

4. **Performances.** The assessment of a model's effectiveness is reliant on four distinct metrics. Accuracy evaluates the model's ability to generate correct output, while specificity or precision measures the model's ability to accurately detect negative outcomes. Sensitivity or recall, on the other hand, evaluates the model's performance in accurately

identifying positive outcomes. The F1-score provides a comprehensive evaluation of the model's performance, by considering both precision and recall.

In Table 1, we show the outcomes of our study, which is based on three datasets: Dataset 1 (ImVia), Dataset 2 (URFD), and Dataset 3 (FallAllD). Our findings indicate that we achieved an average classification accuracy of 96.66%, along with 89.95% and 96.72% sensitivity and specificity, respectively. The average accuracy is determined by adding the correctly classified values (true positive) and dividing it by the total number of values, resulting in 96.66%.

Table 1. Performance

Dataset	Precision (%)	Recall (%)	F1-score (%)	Accuracy (%)
ImVia	95.90	86.89	90.95	96.48
URFD	97.43	91.16	93.90	95.65
FallAllD	100.0	100.0	100.0	100.0
Overall	96.72	89.95	93.08	96.66

Table 2 presents the accuracy results of the fall detection system obtained from testing the proposed method and compared to four other state-of-the-art computer vision methods. The evaluation was conducted using the same datasets employed in our research paper. The table demonstrates that the proposed method achieves the highest level of accuracy in both URFD and overall assessments, indicating its superior performance compared to the other methods.

Table 2. Comparison between ours and other related approaches

Research	Methods	Accuracy (%)			
		URFD	FallAllD	ImVia	Overall
Núñez-Marcos et al. [32]	Optical Flow+CNN-FC-NN	95.00	–	–	–
Menacho and Ordonez [33]	Optical Flow+CNN	88.55	–	–	–
Xu et al. [34]	OpenPose+CNN	*	–	–	91.70
Namburi and Hengsanankun [3]	OpenPose+SVM	93.77	92.82	90.28	89.66
Proposed method	BlazePose+CNN-LSTM	95.65	100.00	96.48	96.66

*: The study incorporates the URFD Dataset along with two supplementary datasets. However, it exclusively reports the overall accuracy and does not furnish individual accuracy scores for each dataset.

5. **Conclusion and Future Works.** In conclusion, we used a vision-based method to reduce discomfort from wearing, the possibility of forgetting to wear tracking devices, and the need to change batteries frequently. We extracted four skeleton points from BlazePose and calculated angles, linear velocity, and angular velocity to create time-series data similar to gyroscopes and accelerometers. We then fed these data into a CNN-LSTM to detect falls. This method achieved high accuracy (96.66%), precision (96.72%), and recall (89.95%). However, although our approach can reduce fault alarms caused by fast body movements that resemble falls, there are limitations. The skeleton data extracted from BlazePose has more errors than sensors, and falls are dangerous events that can be life-threatening. Therefore, our method may need to be combined with other fall detection methods to increase accuracy, such as using multiple cameras to reduce errors caused by object occlusion or angles that BlazePose cannot detect all 33 landmarks accurately. Alternatively, combining sensor-based and vision-based approaches may yield satisfactory results and be suitable for real-world applications. This method is suitable as a backup

system for fall detection in older people who may forget to wear tracking devices or experience discomfort wearing them and in public places to reduce losses and provide timely assistance.

Acknowledgment. This study received support from the Faculty of Engineering at Mahasakham University and the Faculty of Science and Engineering at Kasetsart University. The authors express their gratitude to the reviewers for their valuable feedback and suggestions that have enhanced the quality of this work.

REFERENCES

[1] *Public Health Statistics*, http://www.thaincd.com/2016/mission/documents-detail.php?id=13373&tid=39&gid=1-027, Accessed on February 19, 2023.

[2] World Health Organization, *Falls*, https://www.who.int/news-room/fact-sheets/detail/falls, Accessed on May 22, 2021.

[3] A. Namburi and T. Hengsanankun, Combining SVM and human-pose for a vision-based fall detection, *ICIC Express Letters, Part B: Applications*, vol.13, no.11, pp.1177-1187, 2022.

[4] Y. Wang, W. Xiong, J. Yang and S. Wang, A new fall detection method based on fuzzy reasoning for an omni-directional walking training robot, *International Journal of Innovative Computing, Information and Control*, vol.16, no.2, pp.597-608, 2020.

[5] Z. Chen and Y. Wang, Infrared ultrasonic sensor fusion for support vector machine based fall detection, *Journal of Intelligent Material Systems and Structures*, vol.29, no.9, pp.2027-2039, 2018.

[6] Y. Wu, Y. Su, R. Feng, N. Yu and X. Zang, Wearable-sensor-based pre-impact fall detection system with a hierarchical classifier, *Measurement*, vol.140, pp.283-292, 2019.

[7] S.-L. Hsieh, C.-C. Chen, S.-H. Wu and T.-W. Yue, A wrist-worn fall detection system using accelerometers and gyroscopes, *Proc. of the 11th IEEE International Conference on Networking, Sensing and Control*, pp.518-523, 2014.

[8] W. Lu, M. C. Stevens, C. Wang, S. J. Redmond and N. H. Lovell, Smart triggering of the barometer in a fall detector using a semi-permeable membrane, *IEEE Transactions on Biomedical Engineering*, vol.67, no.1, pp.146-157, 2020.

[9] A. Salarian, H. Russmann, F. J. G. Vingerhoets, C. Dehollain, Y. Blanc, P. R. Burkhard and K. Aminian, Gait assessment in Parkinson's disease: Toward an ambulatory system for long-term monitoring, *IEEE Transactions on Biomedical Engineering*, vol.51, no.8, pp.1434-1443, 2004.

[10] J. Klenk, C. Becker, F. Lieken, S. Nicolai, W. Maetzler, W. Alt, W. Zijlstra, J. M. Hausdorff, R. C. van Lummel, L. Chiari and U. Lindemann, Comparison of acceleration signals of simulated and real-world backward falls, *Medical Engineering & Physics*, vol.33, no.3, pp.368-373, 2011.

[11] M. T. Pourazad, A. Shojaei-Hashemi, P. Nasiopoulos, M. Azimi, M. Mak, J. Grace, D. Jung and T. Bains, A non-intrusive deep learning based fall detection scheme using video cameras, *2020 International Conference on Information Networking (ICOIN)*, pp.443-446, 2020.

[12] B. S. Daga, A. A. Ghatol and V. M. Thakare, Silhouette based human fall detection using multimodal classifiers for content based video retrieval systems, *2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)*, pp.1409-1416, 2017.

[13] G. Mastorakis and D. Makris, Fall detection system using Kinect's infrared sensor, *Journal of Real-Time Image Processing*, vol.9, pp.635-646, 2014.

[14] W. Min, S. Zou and J. Li, Human fall detection using normalized shape aspect ratio, *Multimedia Tools and Applications*, vol.78, pp.14331-14353, 2019.

[15] C. Rougier, J. Meunier, A. St-Arnaud and J. Rousseau, Robust video surveillance for fall detection based on human shape deformation, *IEEE Transactions on Circuits and Systems for Video Technology*, vol.21, no.5, pp.611-622, 2011.

[16] M. Yu, S. M. Naqvi and J. Chambers, Fall detection in the elderly by head tracking, *2009 IEEE/SP 15th Workshop on Statistical Signal Processing*, pp.357-360, 2009.

[17] N. Lu, Y. Wu, L. Feng and J. Song, Deep learning for fall detection: Three-dimensional CNN combined with LSTM on video kinematic data, *IEEE Journal of Biomedical and Health Informatics*, vol.23, no.1, pp.314-323, 2019.

[18] X. Zi, K. Chaturvedi, A. Braytee, J. Li and M. Prasad, Detecting human falls in poor lighting: Object detection and tracking approach for indoor safety, *Electronics*, vol.12, no.5, 1259, 2023.

[19] Z. Zhang, C. Conly and V. Athitsos, Evaluating depth-based computer vision methods for fall detection under occlusions, in *Advances in Visual Computing. ISVC 2014. Lecture Notes in Computer Science*, G. Bebis et al. (eds.), Cham, Springer, 2014.

[20] S. Denkovski, S. S. Khan, B. Malamis, S. Y. Moon, B. Ye and A. Mihailidis, Multi visual modality fall detection dataset, *IEEE Access*, vol.10, pp.106422-106435, 2022.
[21] M. Kepski and B. Kwolek, Fall detection using ceiling-mounted 3D depth camera, *2014 International Conference on Computer Vision Theory and Applications (VISAPP)*, vol.2, pp.640-647, 2014.
[22] Y.-H. Nho, S. Ryu and D.-S. Kwon, UI-GAN: Generative adversarial network-based anomaly detection using user initial information for wearable devices, *IEEE Sensors Journal*, vol.21, no.8, pp.9949-9958, 2021.
[23] I. Grishchenko, V. Bazarevsky, A. Zanfir, E. G. Bazavan, M. Zanfir, R. Yee, K. Raveendran, M. Zhdanovich, M. Grundmann and C. Sminchisescu, BlazePose GHUM Holistic: Real-time 3D human landmarks and pose estimation, *arXiv Preprint*, arXiv: 2206.11678, 2022.
[24] ImVia Laboratory, *Fall Detection Dataset*, https://imvia.u-bourgogne.fr/en/database/fall-detection-dataset-2.html, Accessed on May 22, 2021.
[25] B. Kwolek and M. Kepski, Human fall detection on embedded platform using depth maps and wireless accelerometer, *Computer Methods and Programs in Biomedicine*, vol.117, pp.489-501, 2014.
[26] M. Saleh, M. Abbas and R. L. B. Jeannès, FallAllD: An open dataset of human falls and activities of daily living for classical and deep learning applications, *IEEE Sensors Journal*, DOI: 10.1109/JSEN.2020.3018335, 2020.
[27] R. Hughey and A. Krogh, Hidden Markov models for sequence analysis: Extension and analysis of the basic method, *Bioinformatics*, vol.12, no.2, pp.95-107, 1996.
[28] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár and C. L. Zitnick, Microsoft COCO: Common objects in context, in *Computer Vision – ECCV 2014. ECCV 2014. Lecture Notes in Computer Science*, D. Fleet, T. Pajdla, B. Schiele and T. Tuytelaars (eds.), Cham, Springer, 2014.
[29] G. Jocher, K. Nishimura, T. Mineeva and R. Vilarino, YOLOv5 (2020),*GitHub Repository*, https://github.com/ultralytics/yolov5, Aocessed on May 22, 2022.
[30] N. Wojke, A. Bewley and D. Paulus, Simple online and realtime tracking with a deep association metric, *2017 IEEE International Conference on Image Processing (ICIP)*, pp.3645-3649, 2017.
[31] G. Welch, G. Bishop et al., *An Introduction to the Kalman Filter*, Technical Report, University of North Carolina at Chapel Hill, Chapel Hill, NC, United States, 1995.
[32] A. Núñez-Marcos, G. Azkune and I. Arganda-Carreras, Vision-based fall detection with convolutional neural networks, *Wireless Communications and Mobile Computing*, 2017.
[33] C. Menacho and J. Ordonez, Fall detection based on CNN models implemented on a mobile robot, *2020 17th International Conference on Ubiquitous Robots (UR)*, pp.284-289, 2020.
[34] Q. Xu, G. Huang, M. Yu and Y. Guo, Fall prediction based on key points of human bones, *Physica A: Statistical Mechanics and Its Applications*, vol.540, 123205, 2020.