Advances in Science, Technology and Engineering Systems Journal Vol. 9, No. 6, 12-20 (2024)

www.astesj.com

Special Issue on Innovation In Computing, Engineering Science & Technology

ASTESJ

ISSN: 2415-6698

Advanced Fall Analysis for Elderly Monitoring Using Feature Fusion and CNN-LSTM: A Multi-Camera Approach

Win Pa Pa San[1], Myo Khaing[2]

[1]*Image and Signal Processing Lab, University of Computer Studies, Mandalay, Mandalay, 05071, Myanmar*

[2]*Faculty of Computer Science, University of Computer Studies, Mandalay, Mandalay, 05071, Myanmar*

ARTICLE INFO

Article history:
Received: 15th September, 2024
Revised: 30th September, 2024
Accepted: 15th October, 2024
Online: 30th November, 2024

Keywords:
Feature Fusion
Human Silhouette Image (HSI)
Silhouette History Images (SHI)
Dense Optical Flow (DOF)
Convolutional Neural Network (CNN)
Long Short-Term Memory (LSTM)

ABSTRACT

As society ages, the imbalance between family caregivers and elderly individuals increases, leading to inadequate support for seniors in many regions. This situation has ignited interest in automatic health monitoring systems, particularly in fall detection, due to the significant health risks that falls pose to older adults. This research presents a vision-based fall detection system that employs computer vision and deep learning to improve elderly care. Traditional systems often struggle to accurately detect falls from various camera angles, as they typically rely on static assessments of body posture. To tackle this challenge, we implement a feature fusion strategy within a deep learning framework to enhance detection accuracy across diverse perspectives. The process begins by generating a Human Silhouette Image (HSI) through background subtraction. By combining silhouette images from two consecutive frames, we create a Silhouette History Image (SHI), which captures the shape features of the individual. Simultaneously, Dense Optical Flow (DOF) extracts motion features from the same frames, allowing us to merge these with the SHI for a comprehensive input image. This fused representation is then processed using a pre-trained Convolutional Neural Network (CNN) to extract deep features. A Long Short-Term Memory (LSTM) Recurrent Neural Network (RNN) is subsequently trained on these features to recognize patterns indicative of fall events. Our approach's effectiveness is validated through experiments on the UP-fall detection dataset, which includes 1,122 action videos and achieves an impressive 99% accuracy in fall detection.

1. Introduction

The aging population is rapidly growing worldwide, leading to a significant increase in the number of elderly individuals who require constant care and monitoring. As a result, the ratio of family caregivers to elderly individuals is becoming increasingly unbalanced, especially in countries with higher life expectancies. This imbalance has created a pressing need for automatic health monitoring systems that can provide timely and efficient care for the elderly. One of the most critical aspects of such health monitoring systems is the detection of falls, a leading cause of injury and hospitalization among older adults.

Falls among the elderly can occur for various reasons, including heart attacks, high blood pressure, and other home accidents. The consequences of falls can be severe, often leading to a decline in physical and mental health, reduced mobility, and increased dependence on caregivers. Therefore, accurately detecting falls in real-time is essential for preventing further injuries and ensuring prompt medical attention. Despite the importance of fall detection, traditional vision-based systems face significant challenges in achieving reliable performance across different environments and camera viewpoints.

In recent years, computer vision and machine learning have paved the way for more sophisticated fall detection systems. Convolutional Neural Networks (CNNs) have shown remarkable success in various image processing and object recognition tasks, making them suitable candidates for analyzing video data in fall detection applications. However, static image-based approaches often struggle to capture the temporal dynamics of fall events, which are crucial for accurate detection. This limitation can be addressed by integrating Recurrent Neural Networks (RNNs), particularly Long Short-Term Memory (LSTM) networks, which excel at learning temporal dependencies in sequential data.

.* Win Pa Pa San, University of Computer Studies, Mandalay, Mandalay, 05071, Myanmar, +959262988945, winpapasan@ucsm.edu.mm

https://dx.doi.org/10.25046/aj090602

The proposed fall detection system leverages the strengths of both CNNs and LSTMs, combined with a feature fusion approach to enhance the accuracy and robustness of fall detection. The system utilizes multiple cameras to capture different viewpoints of the monitoring area, providing a comprehensive view of the scene. Human silhouette images are extracted from two consecutive video frames and fused into a Silhouette History Image (SHI), which serves as a shape feature representing the subject's posture over time. Additionally, Dense Optical Flow (DOF) is computed to capture motion features between frames, offering valuable information about the subject's movements.

By fusing SHI and DOF features, the system creates a rich representation of both spatial and temporal aspects of the scene. These fused features are then fed into a pre-trained CNN to extract deep features, which are subsequently processed by an LSTM network to recognize fall events. The use of multiple cameras ensures that the system can detect falls from various angles, overcoming the limitations of single-camera setups. Furthermore, the feature fusion approach enables the system to capture subtle changes in posture and movement, improving the overall detection accuracy.

To evaluate the effectiveness of the proposed system, experiments were conducted using the publicly available UP-Fall detection dataset. The results demonstrate that the proposed method outperforms traditional vision-based fall detection systems, achieving superior performance in terms of accuracy and robustness. This research highlights the potential of combining feature fusion with CNN-LSTM architectures for developing advanced fall detection systems that can significantly enhance the safety and well-being of elderly individuals.

The primary aim of this research is to develop an advanced fall detection system that accurately identifies fall events in real-time, leveraging feature fusion and CNN-LSTM architectures within a multi-camera setup. The specific objectives are:

To design a robust fall detection framework that integrates shape and motion features using Silhouette History Images (SHI) and Dense Optical Flow (DOF).

- To employ a pre-trained CNN for deep feature extraction and an LSTM network for temporal sequence analysis to improve fall detection accuracy.
- To validate the effectiveness of the proposed system through extensive experiments using a publicly available dataset, ensuring its practical applicability in various indoor environments.

The motivation for this research stems from the growing need for reliable and efficient fall detection systems in elderly care. With the increasing elderly population, there is a heightened demand for solutions that can monitor and ensure the safety of older adults, particularly those living alone or in assisted living facilities. Existing fall detection systems often struggle with accuracy due to limitations in capturing dynamic movements and variations in camera viewpoints. By addressing these challenges through the integration of advanced machine learning techniques and a multi-camera approach, this research aims to provide a more dependable solution that enhances the quality of life for the elderly.

Traditional vision-based fall detection systems face several challenges, including:

- Inability to capture temporal dynamics of fall events, leading to missed detections or false alarms.
- Limited performance due to reliance on single-camera setups, which cannot cover all angles and may result in occlusions.
- Difficulty in accurately distinguishing between falls and other similar activities, such as sitting down abruptly.

The proposed system combines CNN and LSTM networks to leverage their strengths in spatial and temporal feature extraction. The use of multiple cameras ensures comprehensive coverage of the monitored area, reducing the likelihood of occlusions and improving detection reliability. Feature fusion of SHI and DOF provides a rich representation of both posture and movement, enabling the system to differentiate between falls and non-fall activities more accurately.

This research makes several key contributions to the field of fall detection:

- Introduction of a novel feature fusion approach that combines SHI and DOF to capture both shape and motion characteristics of potential fall events.
- Development of a hybrid CNN-LSTM architecture that effectively integrates spatial and temporal features for enhanced fall detection performance.
- Implementation of a multi-camera system that overcomes the limitations of single-camera setups, providing a more robust and reliable solution for real-world applications.
- Extensive experimental validation using the UP-Fall detection dataset, demonstrating the superior accuracy and robustness of the proposed method compared to traditional systems.

By addressing the limitations of existing fall detection approaches and introducing innovative solutions, this research contributes to the advancement of health monitoring technologies, ultimately improving the safety and well-being of elderly individuals. Moreover, the proposed system can be applied to a smart home system to assist and provide telehealth services for the elderly.

This paper is organized as follows. Section I describes the objectives, motivations, system problem with solution, and contribution of this study. The literature survey about various fall detections is analyzed in Section II. The system overview and the detailed explanation of this study are presented and the experimental results and comparison with the results of the other existing methods are presented in Section III. Some discussion about the pros and cons of the proposed system are discussed in Section IV. Finally, the conclusion and future work are drawn in Section V.

2. Related Work

The advancement of sophisticated sensors and devices has captured the interest of many researchers focused on artificial intelligence systems. This is particularly true for applications such as smart home systems, patient monitoring, surveillance, and elderly monitoring, where various sensor-based and camera-based approaches have been proposed. Fall detection systems, in particular, can be classified into two categories based on the sensors used: sensor-based and camera (vision)-based.

2.1. Sensor-based Fall detection

Fall detection sensors typically incorporate accelerometers and gyroscopes to monitor the acceleration and orientation of elderly individuals. When attached to various body parts, accelerometers collect acceleration data during falls. One proposed system [1] employs accelerometers and gyroscopes mounted on the gait to assess balance, detect falls, and evaluate fall risk. In a different approach, Lindeman et al. integrated accelerometer sensors into a hearing aid positioned behind the ear [2]. Another fall detection system [3] identifies falls and locates the fallen individual. This system utilizes a sensor attached to the waist to detect backward and sideways falls based on the wearer's final orientation.

Additionally, the authors in [4] developed a machine learning-based fall detection system that utilizes temporal and magnitude features extracted from acceleration signals. These features were used to train a Support Vector Machine classifier for fall identification. Bianchi et al. implemented a fall detection system using barometric pressure sensors, evaluating its performance against accelerometer-based systems; this system classifies falls based on postural orientation and altitude changes [5]. In [6], another system was proposed that not only detects falls but also assesses injury severity, employing multiple accelerometers attached to joints to analyze three-axis acceleration data. Furthermore, in [7], the authors introduced a fall detection system that combines accelerometer sensors with the Discrete Wavelet Transform (DWT) and Support Vector Machine (SVM) algorithm.

2.2. Vision-based Fall detection

Numerous fall detection systems have been developed in recent years, each utilizing different techniques to enhance accuracy and reliability. A notable approach employs key points of the human skeleton detected via OpenPose, as demonstrated in [8]. This system identifies falls based on the speed of descent of the hip joint, the centerline angle, and the body's width-to-height ratio. While it achieves 98.3% sensitivity, 95% specificity, and 97% accuracy on a dataset of 60 falling and 40 non-falling actions, the system encounters challenges with partial occlusion and recognizing falls from multiple directions.

Another vision-based approach for fall detection, utilizing multiple cameras and convolutional neural networks (CNNs), was proposed in [9]. This system leverages optical flow to capture relative motion between consecutive images and trains three CNN models to process visual features from different camera angles. The results on the UP-Fall detection dataset demonstrated 95.64% accuracy, 97.95% sensitivity, and 83.08% specificity. However, the system's performance is impacted by environmental changes and occlusions. In [10], the authors developed a fall detection system that employs features extracted by Inception v3 and a MobileNet model for human detection. By applying transfer learning, they achieved 98.5% accuracy, 97.2% specificity, and 93.47% sensitivity on the FDD dataset, and 91.5% accuracy, 94% specificity, and 100% sensitivity on the URFD dataset. Nonetheless, managing occlusions continues to pose a significant challenge.

Similarly, in [11], the authors proposed a vision-based fall detection method using CNNs, which involved a three-step training process: initial training with ImageNet, motion modeling with UCF101, and fine-tuning specifically for fall detection. Testing on the URFD, Multicam, and FDD datasets resulted in accuracy rates of 95%, 96%, and 97%, respectively. While the results are promising, the system requires improvements in avoiding image preprocessing issues and managing occlusions and multi-person detection. In [12], the authors combined histograms of oriented gradients (HOG), local binary patterns (LBP), and Caffe features for fall detection. Their system utilized VIBE+ for human detection and extraction, along with SVM for classification, achieving sensitivities of 95%, 93.3%, and 92.9%, and specificities of 97.5%, 92.2%, and 86.4% on the Multicam, Chua's dataset, and their dataset, respectively. However, handling occlusions remains a challenge.

Furthermore, in [13], the authors focused on detecting fallen individuals using assistive robots. Their system utilized features such as the aspect ratio of the bounding box, normalized bounding box width, and bottom coordinate, employing an SVM-based classifier. Testing on the FPDS dataset yielded 100% precision and 99.74% recall. However, the system requires enhancements in occlusion detection and minimizing image preprocessing issues.

These studies underscore several common challenges fall detection systems face, including occlusion handling, adaptability to diverse environmental conditions, effective feature extraction and fusion, thorough testing across varied datasets, and detecting falls in multi-person environments. The proposed advanced fall detection system aims to tackle these issues by integrating shape and motion features, utilizing a hybrid CNN-LSTM architecture, and employing a multi-camera setup. This approach promises to enhance the accuracy and reliability of fall detection, making significant progress toward robust and practical real-world applications.

3. Material and Methods

The purpose system flow of the block diagram illustrating the system flow is shown in Figure. 1 of the Advanced Fall Detection System Using Feature Fusion and CNN-LSTM. They are:

- Video Input: Multiple camera feeds provide input data capturing the indoor environment from different viewpoints.
- Data Preprocessing: Initial processing steps such as frame rate adjustment and background subtraction are performed to prepare the input data for feature extraction.
- Feature Extraction: Shape and motion features are extracted from the preprocessed video frames, capturing relevant information about human postures and movements.
- Feature Fusion: The extracted shape features (SHI) and motion features (DOF) are fused into a unified feature representation, combining both the spatial and temporal information.
- CNN-LSTM: The fused features are input to a hybrid CNN-LSTM architecture, where CNN layers extract spatial features, and LSTM layers model temporal dependencies across frames.

- Fall Detection: The learned features are used for fall event detection, where thresholding and event recognition techniques are applied to identify fall events within the video sequences.
- Classification Output: The system outputs the results of fall event detection, indicating the presence or absence of fall events in the monitored environment.

In this system, the sequential flow of data and processing steps in the fall detection system: In the first step, the fall detection system utilizes multiple camera feeds to capture the indoor environment from diverse viewpoints. These camera feeds serve as the primary input data for the system, providing comprehensive coverage of the monitored area. Before further processing, initial preprocessing steps are conducted to ensure the input data is suitable for feature extraction. This includes adjustments to the frame rate of the video streams to optimize computational efficiency and standard background subtraction techniques to segment foreground objects from the static background.

In the second step the following preprocessing, the system extracts shape and motion features from the preprocessed video frames. Shape features are derived from human silhouette images obtained through background subtraction, while motion features are computed using dense optical flow techniques applied to consecutive frames. These features capture essential information regarding human postures and movements within the monitored environment, serving as discriminative cues for fall event detection.

In the third step, the extracted shape and motion features are fused into a unified feature representation using a feature fusion approach. This fusion process combines spatial and temporal information, leveraging the complementary nature of shape and motion cues to enhance the discriminative power of the feature representation. The fused features, called Silhouette History Image (SHI) and Dense Optical Flow (DOF) Image, respectively, from the input data for subsequent processing stages.

In the fourth step, the fused features are input to a hybrid CNN-LSTM architecture, designed to capture spatial and temporal dependencies within the input data effectively. The CNN component of the architecture extracts spatial features from the fused representations, leveraging convolutional layers to learn hierarchical representations of the input features. These spatial features are then fed into LSTM layers, which model temporal dynamics across consecutive frames, allowing the system to capture the sequential nature of human actions and movements.

In the fifth step, the learned features from the CNN-LSTM architecture are utilized for fall event detection within the video sequences. This involves applying thresholding and event recognition techniques to the learned representations, enabling the system to identify instances of fall events based on predefined criteria. The combination of spatial and temporal features, along with the robust architecture of the CNN-LSTM model, facilitates accurate and reliable fall detection performance.

Finally, the system outputs the results of fall event detection, indicating the presence or absence of fall events in the monitored environment. These results provide valuable insights into the safety and well-being of individuals within the indoor space, enabling timely intervention and assistance in the event of a fall.

Background subtraction is a critical preprocessing step in the fall detection system, aimed at isolating human subjects from the static background in the video feeds. This process involves several stages to accurately detect and segment the moving foreground objects, which is essential for subsequent feature extraction and analysis.

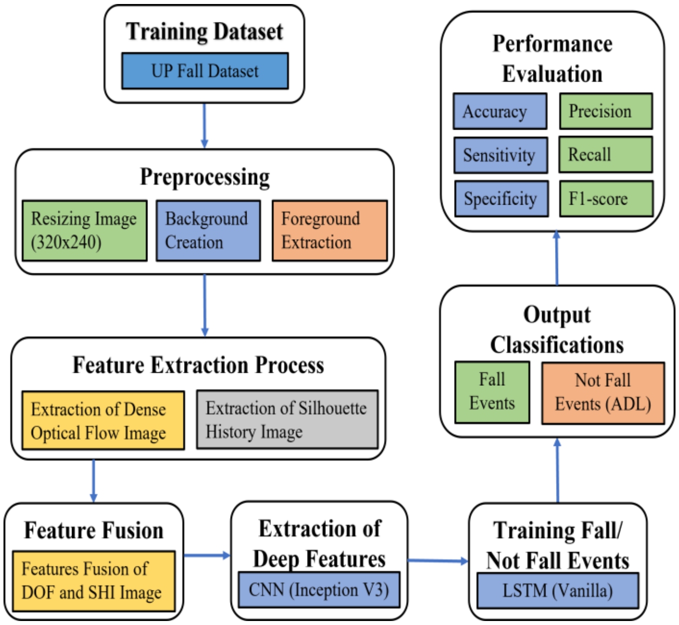

Figure 1: System flow of the advanced fall detection system using feature fusion and CNN-LSTM

3.1. Preprocessing

A) Background Creation

The first step in background subtraction is to create a background frame that represents the static elements in the scene. This is particularly challenging in fall detection scenarios where the human subject is often present throughout the video. Traditional methods like Gaussian Mixture Models (GMM) are inadequate in such cases due to their inability to handle the continuous presence of the subject. Instead, we employ a method based on frame differencing and foreground replacement:

(1) Common Background Frame (CBF) Selection: Identify a frame from the video sequence that does not contain any moving objects or humans. This frame is used as the CBF.

(2) Foreground Replacing: For videos without a clear background frame, the following steps are performed:

- Human Segmentation Mask (M): Utilize Mask-RCNN to generate a segmentation mask for the human subject.
- Pixel Replacement: Replace the pixels in the mask (M) with the corresponding pixels from the CBF using the equation:

$$BF(x,y) = \begin{cases} CBF(x,y) & if\ M(x,y)\ = 0 \\ F(x,y) & if\ M(x,y) = 1 \end{cases} \quad (1)$$

- Background Frame (BF) Storage: Save the resulting frame as the background frame for the video sequence.

B) Foreground Extraction

Once the background frame (BF) is established, the next step is to extract the foreground objects. This involves comparing each frame (F) of the video to the background frame to identify moving objects:

$$FG(x,y) = \begin{cases} 1 & if\ \ BF(x,y) - F(x,y) \geq TH \\ 0 & otherwise \end{cases} \qquad (2)$$

The threshold (*TH*) is the pixel value that can differentiate the moving foreground and background objects. The illustration of the process of foreground extraction results is shown in Figure. 2.

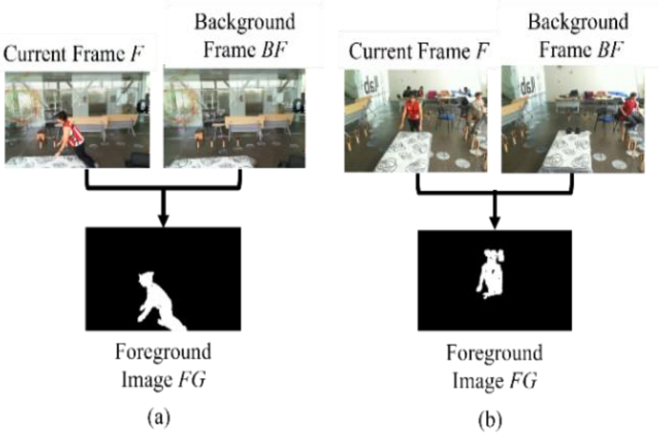

Figure 2: Illustration of foreground extraction results from (a) camera1 and (b) camera2

C) Noise Removing

After extracting the foreground, it is essential to filter out noise and ensure only the relevant human subjects are retained:

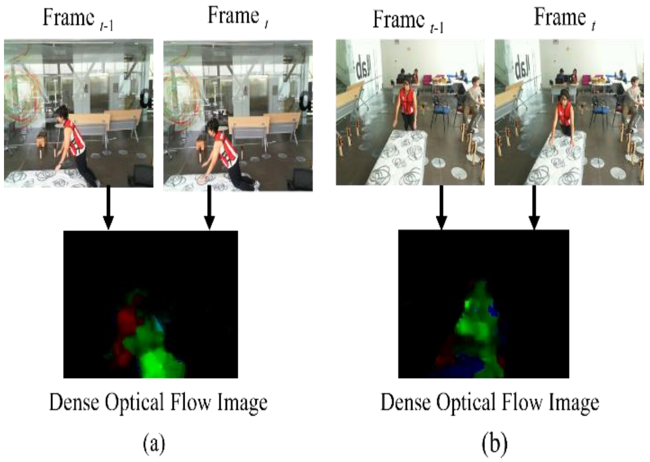

Figure 3: Background subtraction results (1st row: input frames, 2nd row: foreground

(1) Object Classification: Analyze the foreground mask to identify the human subject acting. Non-human objects are considered noise.

(2) Noise Filtering: Apply size-based filtering and morphological operations to remove small, irrelevant objects from the foreground mask. This step ensures that only the significant moving objects (humans) are retained for further processing. Some more sample images of background subtraction results are shown in Figure. 3.

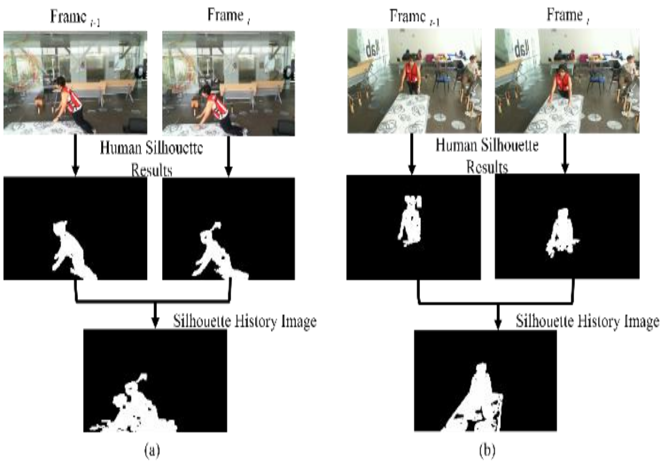

Figure 4: Creation of silhouette history image (SHI) (a) camera1 (b) camera2

3.2. Feature Extraction

A) Extraction of Shape Feature

To extract the shape feature, the edge smoothing process is performed over the noise-removed human silhouette image (foreground results). Then the resulting human silhouette images of two consecutive frames are combined to create the Silhouette History Image (SHI) results, which are used as the shape features, as shown in Figure. 4.

B) Extraction of Motion Feature

Dense optical flow calculation [14] is used for motion feature extraction. Dense optical flow features are extracted from every two consecutive frames. Colors are then assigned to the dense optical flow results using the HSV color space. The orientation value calculated from the dense optical flow is assigned as the Hue value, the Saturation is set to the maximum of 255, and the magnitude value of the dense optical flow is assigned as the Value in the HSV color space. The results of motion feature extraction from Camera1 and Camera2 are shown in Figure. 5 (a) and (b).

3.3. Feature Fusion

In this part, SHI and DOF are fused into a single input data for the training model. SHI and DOF have the same image size, and feature fusion (*FF*) is performed using the following equation. The fused feature dimensions will be the same as those of the original input images with 320×240 image size, and the result of feature fusion is shown in Figure. 6.

$$FF(x,y) = \begin{cases} SHI(x,y) & if\ SHI(x,y) = 1\ , DOF(x,y) = 0 \\ DOF(x,y) & otherwise \end{cases} \qquad (3)$$

3.4. Train CNN-LSTM for Fall Detection

A) Extraction of Deep Features using Convolutional Neural Network (CNN)

A convolutional neural network (CNN) is an artificial neural network designed to process image data and learn to classify and segment various objects within images and videos. The Inception V3 model, known for its effectiveness in image analysis and object detection, is utilized in the system to extract deep features from the input image fusion data. Inception V3, a third edition of Google's Inception CNN, consists of 42 layers. The output from the average pooling layer, a 2048-dimensional feature vector, is used as the deep features for fall detection.

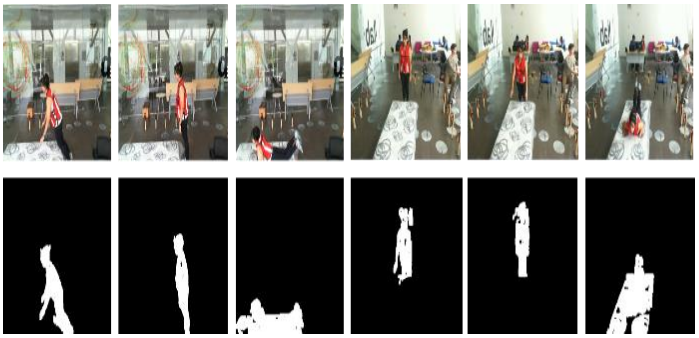

Figure 5: Motion feature extraction results (a) camera1 (b) camera2

B) *Training Fall Event Detection Model using Recurrent Neural Network (RNN)*

A Recurrent Neural Network (RNN) is designed for learning from sequential or time-series data, where the output depends on prior elements in the sequence. In this system, Long Short-Term Memory (LSTM), which consists of a cell, an input gate, an output gate, and a forget gate, is used for detecting fall events.

As shown in Figure. 7, the fused feature outputs from two cameras are fed into the Inception V3 model, pre-trained on the large ImageNet dataset. The "avg-pool" layer of Inception V3 produces a deep feature vector of length 2048. Deep features from both cameras are combined to create a feature vector of length 4096. This feature vector sequence, comprising 18 frames (spanning 3 seconds), is then fed into an LSTM for training to detect whether the input sequences contain a fall event. The LSTM used for fall detection consists of 2 stacked layers with 512 hidden units, as shown in Figure. 8. We used the ReLU activation function in two hidden layers and in the final output layer, softmax is applied for classifying the fall and not-fall events.

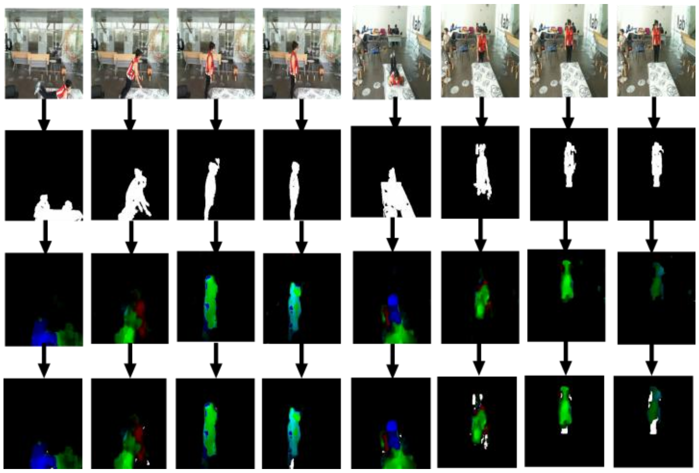

Figure 6: Sample results of feature fusion (1st row: input images, 2nd row: shape feature results, 3rd row: motion feature results, 4th row: feature fusion results)

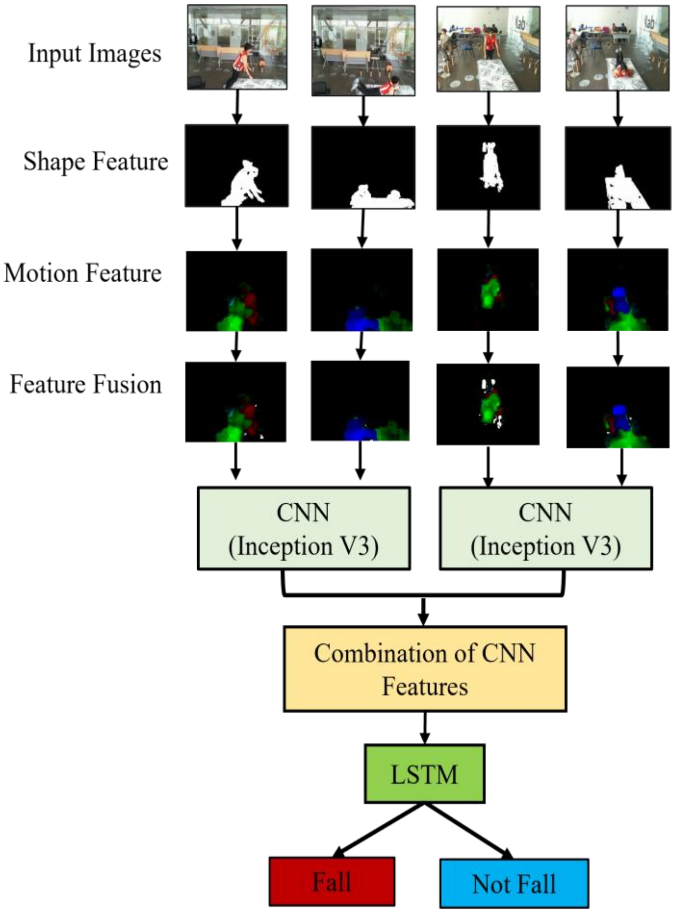

Figure 7: Flow chart of fall detection using CNN-LSTM

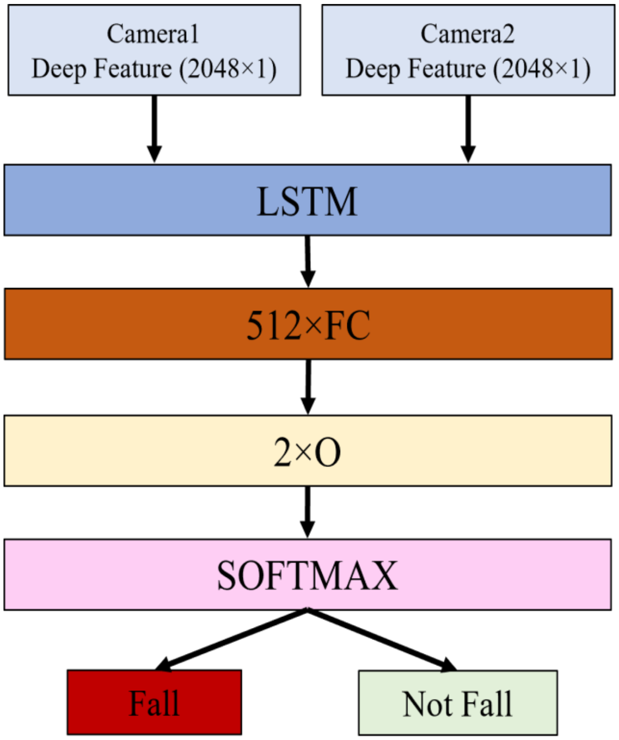

Figure 8: Architecture of fall detection model using CNN-LSTM

4. Experimental Results

4.1. Dataset

The UP-Fall Detection dataset [15], provided by Universidad Panamericana, Mexico in April 2019, includes data from 6 infrared sensors, 6 accelerometers, 3 Raspberry Pi devices, 2 cameras, and 1 brain sensor to create a multimodal dataset for fall detection. This research uses only data from the 2 cameras to implement vision-based fall detection. The dataset contains 1122 videos, each ranging from 10 to 60 seconds in length. These videos comprise 11 activities performed by 17 subjects, each repeated 3 times. Activities 1 to 5 are falls, while the remaining activities are daily living, as detailed in Table I.

The UP-Fall detection dataset provides the action videos with a frame rate of 18 fps. We use the frame rate of 6fps because most fall events take around 2 or 3 secs and according to experiments, 6fps is enough to perform the fall detection. We convert the frame rate of 18 fps into 6 fps by taking every 3rd frame from the image sequence. Then, foreground extraction is applied to 2 cameras, 3 trials, and activity 1 to 11 of all 17 subjects. The resolution of the RGB image is 320×240 and the following are some results of foreground extraction. The experiments are performed on a 2.2GHz Intel Core i7 CPU machine. The features extraction time of SHI and DOF are 0.011 s and 0.031 s respectively. The features fusion and fall detection time (3s video frames) are 0.016 s and 1.5 s respectively using Python. Some test images of the results of falls and others are shown in Figure 9.

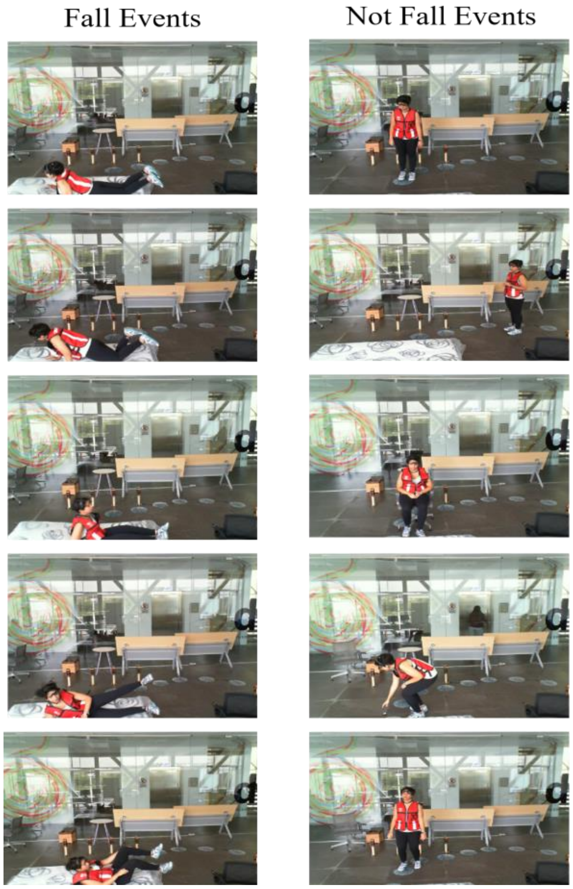

Figure 9: Some test image results of the UP-Fall detection dataset

Table 1: Activities and Their Duration

No.	Activity	Duration (sec)
1	Falling forward using hands	10
2	Falling forward using knees	10
3	Falling backward	10
4	Falling sideward	10
5	Falling while attempting to sit in an empty chair	10
6	Walking	60
7	Standing	60
8	Sitting	60
9	Picking up an object	10
10	Jumping	30
11	Laying	60

4.2. Participants

In the implementation of the advanced fall detection system, we utilized the UP-Fall Detection Dataset [16], which includes 11 activities and three trials per activity. Data were collected from over 17 participants, who were called subjects. Participants performed six simple human daily activities as well as five different types of human falls. During data collection, 17 subjects (9 male and 8 female) ranging from 18–24 years old, mean height of 1.66 m and a mean weight of 66.8 kg, were invited to perform 11 different activities for creating a comprehensive dataset for training and testing the fall detection system. Each participant's data was recorded using multiple modalities, but for this study, we focused solely on the video data captured by two cameras.

- Number of Participants: 17
- Activities: 11 distinct activities (5 fall and 6 daily activities)
- Trials: Each participant performed each activity three times, resulting in multiple video sequences for each activity.

In this research, we train 3 classification models. The first model (CNN-LSTM-2-classes) can classify only two classes such as fall and not-fall events. The second model (CNN-LSTM-7-classes) trained to classify 7 classes: fall events and other activities such as walking, standing, sitting, picking up an object, jumping, and laying. The third model (CNN-LSTM-11-classes) can classify all 11 activities as described in Table. 1.

4.3. Performance Evaluation

For fall detection performance evaluation, we trained and tested the data from the UP-Fall dataset using the same criteria as described in [9]. Data from trials 1 and 2 for 17 subjects were used as the training data, while data from trial 3 were used as the test data. To evaluate the performance of this work, the system uses the following six metrics: Accuracy, Sensitivity, Specificity, Precision, Recall, and F1-score, as given by (4)-(9),[17]. The performance evaluation of the three classification models is described in Table 2.

Moreover, we compare the performance of the proposed system with other approaches as shown in Table 3. We obtained the results of the method in [9] from their paper and used the same evaluation method to compare the results. In Table 3, we can see that our proposed method produces higher accuracy than the method described in [9].

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Numbers\ of\ Predictions} \quad (4)$$

$$Sensitivity = \frac{True\ Positive}{True\ Positive + False\ Negative} \quad (5)$$

$$Specificity = \frac{True\ Negative}{True\ Negative + False\ Positive} \quad (6)$$

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \quad (7)$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \quad (8)$$

$$F1Score = 2 \times \frac{Precision.Sensitivity}{Precision + Sensitivity} \quad (9)$$

5. Discussion and Limitations

The proposed system has some limitations in the computational complexity of training the CNN-LSTM model. It needs to extract deep features using CNN and perform sequence classification using LSTM. But that limitation can be overcome by applying high-performance computing devices such as GPU-machines. Another limitation is the occlusion problem. This applied two cameras for detecting fall events. But sometimes falls can occur in an area which only two cameras cannot cover. Therefore, in the future, we plan to extend this research by applying more cameras and configuring the camera set to cover all areas of the home environment of living alone elderly.

Table 2: Performance Evaluation of Three CNN-LSTM Models (Cam1 &Cam2) on UP-Fall Detection Dataset

Models	CNN-LSTM-2 Classes	CNN-LSTM-7 Classes	CNN-LSTM-11 Classes
Accuracy (%)	99	96	93
Sensitivity (%)	98	94	79
Specificity (%)	98	99	99
Precision (%)	99	94	81
Recall (%)	98	94	79
F1-Score (%)	98	94	80

Table 3: Comparison of Fall Detection Model (CNN-LSTM-2 Classes) performance evaluation on UP-Fall Detection Dataset

Method	Espinosa R, et al [9] (Cam1 &Cam2)	Proposed (Cam1 &Cam2)	Proposed (Cam1)	Proposed (Cam2)
Accuracy (%)	95.64	99	99	99
Sensitivity (%)	97.95	98	96	98
Specificity (%)	83.08	98	96	98
Precision (%)	96.91	99	99	97
Recall (%)	-	98	96	98
F1-Score (%)	97.43	98	97	97

6. Conclusion and Future Works

In this research, a vision-based fall detection system using multiple cameras applying CNN-LSTM has been proposed. The main contribution will be taken on the "features extraction and features fusion from multiple cameras", and the architecture of CNN-LSTM for improving fall detection rate. Based on the experimental results performed on the public dataset of the UP-Fall detection dataset, the proposed system got superior performance over the state-of-the-art methods. This fact points out that the feature fusion approach for CNN-LSTM is very effective and promising for the accurate fall detection system. Limitations such as the computation complexity for training CNN-LSTM can be overcome by using high-performance computing devices. Moreover, the multi-camera approach is more cost-effective than the other multi-sensor approaches, and this research will come as applied science research which can give a lot of benefits to human society. In this research, the experiments are only performed on the UP-Fall detection, a large dataset containing 1122 action videos performed by 17 persons. Then, the proposed method got good performance results on that dataset. In the future, to confirm the effectiveness of this proposed method, we will perform more experiments on other datasets of fall detection.

Conflict of Interest

The authors declare no conflict of interest.

Author Contribution

The major portion of the work presented in this paper was carried out by the first author, Win Pa Pa San, under the supervision of the second author, Myo Khaing. Win Pa Pa San also performed the data analysis, implementation, validation, and preparation of the manuscript.

Acknowledgment

I want to extend special thanks to Dr. Myo Khaing, Professor of the Faculty of Computer Science at the University of Computer Studies, Mandalay (UCSM), and Dr. Sai Maung Maung Zaw, Professor and head of the Faculty of Computer Systems and Technologies at the University of Computer Studies, Mandalay (UCSM), for their continuous guidance, support, and suggestions.

References

[1] Q. Li, J.A. Stankovic, M.A. Hanson, A.T. Barth, J. Lach, G. Zhou, 'Accurate, fast fall detection using gyroscopes and accelerometer-derived posture information', Proceedings - 2009 6th International Workshop on Wearable and Implantable Body Sensor Networks, BSN 2009, (June), 138–143, 2009, doi:10.1109/BSN.2009.46.

[2] Y. Li, K.C. Ho, M. Popescu, 'A microphone array system for automatic fall detection', IEEE Transactions on Biomedical Engineering, **59**(5), 1291–1301, 2012, doi:10.1109/TBME.2012.2186449.

[3] Y. Li, Z. Zeng, M. Popescu, K.C. Ho, 'Acoustic fall detection using a circular microphone array', 2010 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC'10, 2242–2245,

2010, doi:10.1109/IEMBS.2010.5627368.

[4] H. Wang, D. Zhang, Y. Wang, J. Ma, Y. Wang, S. Li, 'RT-Fall: A Real-Time and Contactless Fall Detection System with Commodity WiFi Devices', IEEE Transactions on Mobile Computing, **16**(2), 511–526, 2017, doi:10.1109/TMC.2016.2557795.

[5] F. Bianchi, S.J. Redmond, M.R. Narayanan, S. Cerutti, N.H. Lovell, 'Barometric pressure and triaxial accelerometry-based falls event detection', IEEE Transactions on Neural Systems and Rehabilitation Engineering, **18**(6), 619–627, 2010, doi:10.1109/TNSRE.2010.2070807.

[6] R.K. Shen, C.Y. Yang, V.R.L. Shen, W.C. Chen, 'A Novel Fall Prediction System on Smartphones', IEEE Sensors Journal, **17**(6), 1865–1871, 2017, doi:10.1109/JSEN.2016.2598524.

[7] B. Wójtowicz, A. Dobrowolski, K. Tomczykiewicz, 'Fall detector using discrete wavelet decomposition and SVM classifier', Metrology and Measurement Systems, **22**(2), 303–314, 2015, doi:10.1515/mms-2015-0026.

[8] H.U. Openpose, 'Fall Detection Based on Key Points of', Symmetry, 2020.

[9] R. Espinosa, H. Ponce, S. Gutiérrez, L. Martínez-Villaseñor, J. Brieva, E. Moya-Albor, 'A vision-based approach for fall detection using multiple cameras and convolutional neural networks: A case study using the UP-Fall detection dataset', Computers in Biology and Medicine, **115**, 2019, doi:10.1016/j.compbiomed.2019.103520.

[10] S. Sherin, P.M.T. Student, A.J. Assistant, 'Human Fall Detection using Convolutional Neural Network', International Journal of Engineering Research & Technology, **8**(6), 1368–1372, 2019.

[11] A. Núñez-Marcos, G. Azkune, I. Arganda-Carreras, 'Vision-based fall detection with convolutional neural networks', Wireless Communications and Mobile Computing, **2017**, 2017, doi:10.1155/2017/9474806.

[12] K. Wang, G. Cao, D. Meng, W. Chen, W. Cao, 'Automatic fall detection of human in video using combination of features', Proceedings - 2016 IEEE International Conference on Bioinformatics and Biomedicine, BIBM 2016, 1228–1233, 2017, doi:10.1109/BIBM.2016.7822694.

[13] S. Maldonado-Bascón, C. Iglesias-Iglesias, P. Martín-Martín, S. Lafuente-Arroyo, 'Fallen people detection capabilities using assistive robot', Electronics (Switzerland), **8**(9), 2019, doi:10.3390/electronics8090915.

[14] T. Hassner, C. Liu, 'Dense image correspondences for computer vision', Dense Image Correspondences for Computer Vision, 1–295, 2015, doi:10.1007/978-3-319-23048-1.

[15] L. Martínez-Villaseñor, H. Ponce, J. Brieva, E. Moya-Albor, J. Núñez-Martínez, C. Peñafort-Asturiano, 'Up-fall detection dataset: A multimodal approach', Sensors (Switzerland), **19**(9), 2019, doi:10.3390/s19091988.

[16] L. Martinez-Villasenor, H. Ponce, K. Perez-Daniel, 'Deep learning for multimodal fall detection', Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics, **2019-Octob**, 3422–3429, 2019, doi:10.1109/SMC.2019.8914429.

[17] M. Sokolova, G. Lapalme, 'A systematic analysis of performance measures for classification tasks', Information Processing and Management, **45**(4), 427–437, 2009, doi:10.1016/j.ipm.2009.03.002.