LapseNet: A Hybrid CNN-LSTM Approach for Accurate and Efficient Vision-Based Fall Detection

Shaurya Jain*
Computer Systems Lab
Thomas Jefferson High School
for Science and Technology
Alexandria, United States
2025sjain@tjhsst.edu

Soham Jain*
Computer Systems Lab
Thomas Jefferson High School
for Science and Technology
Alexandria, United States
2025sjain1@tjhsst.edu

Anmol Karan*
Computer Systems Lab
Thomas Jefferson High School
for Science and Technology
Alexandria, United States
2025akaran@tjhsst.edu

*Denotes equal contribution from all authors, with names listed alphabetically

***Abstract*—Falls are a major cause of injury and death among the elderly population, particularly in unsupervised settings where victims often remain unattended for extended periods of time. Such incidents can lead to long-term physical and mental disturbances such as fractures, skin burns, blood loss, and trauma. A reliable and effective fall detection system can ensure that support is provided immediately, improving chances of recovery for victims. A diverse range of fall detection methods have been studied and tested, but most have high false positive rates and limited robustness in real-world scenarios. In this study, we present LapseNet, a hybrid convolutional neural network with long short-term memory to detect falls in indoor settings. We utilized data from four publicly available sources, with a total of 250 videos for training and testing the model, which distinguishes between a) falls and b) activities of daily living. LapseNet achieved a training accuracy of 99.43% and a promising testing and validation accuracy of 100%. These results demonstrate the potential to significantly improve elderly care and safety by enabling timely interventions and reducing the risk of long-term complications from falls.**

***Index Terms*—fall detection, deep learning, CNN, LSTM**

I. Introduction

Collapses among the elderly remains a prominent issue worldwide, as victims are often unable to take immediate safety measures due to resultant immobility. According to the Centers for Disease Control and Prevention [1], 37% of the 14 million adults in the United States who experienced a fall required medical attention from 2012 to 2018. A majority of this group includes senior citizens, with 50% of falls among individuals aged 70 or older resulting in injuries needing medical treatment, according to Vaishya and Vaish [2]. Particularly in the United States, collapses often occur among individuals in isolated places or residents who live alone. Isolated victims of falls, while immobile, are often left helpless for extended periods of time, which can severely damage their mental and physical health.

An effective, lightweight fall detection system would significantly advance efforts to ensure the well-being of the elderly. LapseNet utilizes long short-term memory (LSTM) and a convolutional neural network (CNN) for lightweight and accurate detection of falls. Using other architectures and methods, numerous studies have explored novel approaches to fall detection. For instance, several past fall detection systems have utilized wearable technologies, acoustic signaling, or camera-based detection. However, these systems face variance in efficacy and reliability, which we seek to overcome in this study.

II. Related Work

A. Wearable Devices

Wearable devices have been the leading solution for detecting falls due to their wireless technology and continuous health monitoring capabilities. Chander et al. [3] emphasize the need for wearable stretch sensors in the field of ergonomics because of the devices' potential to detect collapses in the workplace. They propose the use of soft-robotic-stretch (SRS) sensors for fall detection, finding a high correlation between SRS sensor data and 3D motion capture ankle angle kinematics with minimal error (R^2 = 0.854, RMSE = 1.96, MAE = 1.54). These results suggest that SRS sensors could accurately capture ankle joint kinematics on flat and tilted surfaces. Nonetheless, wearable SRS faces limitations due to its sensitivity to stress-induced movements and impact with an object. Hussain et al. [4] also propose a fall-detection system that implements support vector machine, k-nearest neighbors, and random forest. While their findings present an accuracy between 96.82% and 99.80% in recognizing different falling activities, there were challenges in distinguishing between similar fall patterns. Moreover, their F1 score of 67.10 suggests poor model performance and low precision rates.

B. Sound-Based Approaches

In contrast to wearable devices, sound analysis offers a non-intrusive and scalable solution for detecting falls in the elderly population. Kaur et al. [5] utilize Transformer architecture to classify sound input into ”fall” or ”no fall” categories with an accuracy of 0.8673. Their design of an audio Transformer-based deep learning model is beneficial in settings where existing techniques cannot be implemented due to privacy concerns. However, reliance on ambient sound faces challenges in noisy environments and situations where fall sounds are faint or overshadowed by other noises. Khan et al. [6] propose an unsupervised fall detection system that removes interferences from background sound sources using source separation techniques. They use a one-class support vector machine method to distinguish fall from non-fall sounds, achieving 99.28% accuracy with no interference and 92.93% accuracy with a 75% interference level. Since interference hindered performance of the model, there is an inherent need to improve robustness in varied acoustic environments.

C. Biomedical Systems

Biomedical signal-based models are another alternative for automatic fall detection and health monitoring. Hwang et al. [7] developed a system for fall detection in elderly people using a chest-mounted accelerometer, gyroscope, and tilt sensor. They evaluated their system by experimenting on three adults over the age of 26, identifying falls in 119 out of 123 trials, corresponding to a 96.7% accuracy. While their results appear promising, the model lacks testing on the primary target demographic and the experiment has an insufficient number of trials to prove effective in real-world application. Butt et al. [8] utilize electrocardiogram signals from a wearable sensor to signal a fall, while passing a dataset of a collected sequence of images into a fine-tuned pre-trained CNN. While the CNN achieved an optimal accuracy of 98.44%, the collected data contained variation due to differing signals and noise present in the multiple datasets used.

D. Vision-Based Technology

Computer vision-based fall detection methodologies remain the most widely applicable and reliable form of monitoring falls. A study by Maitre et al. [9] employs a CNN in conjunction with an LSTM network, utilizing signals acquired from ultra-wideband (UWB) radars. The detection system achieved its best accuracy of 89% through a train-test split of 70:30 on their own dataset. The study's dataset, however, comprises only five participants. Marcos et al. [10] presents another CNN architecture to classify a sequence of events as either fall or non-fall. The study, trained with the UR dataset, uses an optical flow images generator to compute the motion between consecutive frames in a video. While it achieved a sensitivity of 100% on the UR dataset, the model's specificity reached only 94.86%, indicating the potential for false negatives.

III. Methodology

Source code for the project can be found at:

https://github.com/sjain2025/LapseNet_Fall_Detection.git

A. Data Acquisition

We used four public datasets to train LapseNet, each containing videos of falls and activities of daily living, or ADL (e.g. kneeling, walking, or picking up an object). The datasets that we used include the UR Fall Detection Dataset [11] made by the University of Rzeszow, the Multiple Cameras Fall Dataset [12] by the University of Montreal, the CAUCAFall Dataset [13], and the UBFC Fall Detection Dataset [14]. The UR Fall Detection Dataset consists of 30 falls taken from two cameras and another 40 videos of ADL, for a total of 100 videos. The Multiple Cameras Fall Dataset includes 24 scenarios, out of which 22 contain a fall and two do not. Each scenario includes views from eight different cameras. We used one camera angle video from the first 19 scenarios, and we used videos from all eight camera angles from the last two scenarios, for a total of 35 videos. The CAUCAFall Dataset is structured by ten different subjects, who performed 10 activities each: five were falls and five were ADL. We did not use videos from Subjects 6 and 8 because the videos were in black and white, whereas the rest were in color. Therefore, we used 80 videos in total from the CAUCAFall Dataset. The UBFC Fall Detection Dataset contains videos from falls in six rooms, from which we used five rooms (office, two coffee rooms, two home rooms) and seven videos from each room to maintain balance in the dataset between ADL and fall videos. Hence, we used 35 videos from this dataset.

B. Preprocessing

The CAUCAFall, Multiple Cameras Fall, and UBFC datasets consisted of `.avi` files, and the UR Fall Detection Dataset stored videos as folders of frames (PNGs). We processed the `.avi` files and folders of frames into lists of images representing the videos. From here we condensed the videos into sequences of 30 frames. For example, we would take every fifth frame from a video with 150 frames, condensing it into 30 frames. This process assisted with making the dataset more lightweight and allowed us to capture essential parts of the video while minimizing the influence of irrelevant details. Furthermore, storing the videos in this manner helped the model easily extract relevant details to predict falls. We resized the images to 128×128, leading to slight loss of quality in images but an increase in processing ability. Furthermore, we utilized the TensorFlow API to create and concatenate the datasets using the `tf.data.Dataset` object and the `from_tensor_slices` method. We used batch sizes of 16 sequences per batch.

C. Model

Fig. 1 shows a summary of the model's input and output shapes, as well as the layers' structure. The model that we constructed utilizes a hybrid CNN and LSTM, a type of recurrent neural network, with Keras and TensorFlow backend. This combination of CNN and LSTM allows the model to effectively learn spatial and temporal features. The model starts with a `TimeDistributed` wrapper around the convolutional (`Conv2D`) and pooling (`MaxPooling2D`) layers, which ensure that each video frame is processed independently. Next, the model implements `Dropout` layers to prevent overfitting and ensure generalization in real-world application. After these layers are flattened, the features are passed into an LSTM layer for sequence modeling, which captures temporal dependencies and patterns across the frame sequences. Following dropout regularization and dense layers, sigmoid activation and binary cross-entropy loss functions output a binary result: 0 for no fall and 1 for fall.

Fig. 2 shows the model's architecture, composed of a 64-cell LSTM, where each cell takes input from a convolutional block. The model processes input video frames using multiple parallel streams, each containing a sequence of layers: `Conv2D` layers for feature extraction, followed by `MaxPooling2D` layers for downsampling, and `Dropout` layers for regularization. This architecture leverages the spatial feature extraction capabilities of CNNs and the sequence modeling strengths of LSTMs to achieve high performance in video classification tasks.

D. Training

Our dataset consists of 250 videos featuring both fall and ADL scenarios. We employed a random split of 70% for training (175 videos), 20% for testing (50 videos), and 10% for validation (25 videos). We trained the model with 83.5 GB of RAM and on a Google Colaboratory A100 GPU, which is ideal for accelerated computing tasks because of its faster training times and high throughput. We conducted training over 80 epochs while using a learning rate of 0.001 and a batch size of 16. Throughout its training, the model utilized the Adam optimizer: an iterative optimization algorithm that minimizes the loss function. After preprocessing and data preparation, the model completed its 80-epoch training in approximately 289.94 seconds, demonstrating its lightweight capabilities.

The model was trained to minimize binary cross-entropy loss on the training set, given by the following equation:

$$\mathcal{L}_{\text{BCE}} = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log(p_i) + (1-y_i)\log(1-p_i)\right] \tag{1}$$

In (1), $\mathcal{L}_{\text{BCE}}$ is binary cross-entropy loss, N is the number of samples in the training set, y_i is the true label (0 or 1) of the i-th sample, and p_i is the predicted probability of the sample being class 1 (fall).

IV. Results and Discussion

We recorded five performance metrics for LapseNet: accuracy, loss, AUC, precision, and recall.

Accuracy measures the overall correctness of the model's predictions across all classes, given by the following equation:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

AUC represents the area under the receiver operating characteristic (ROC) curve, which plots the true positive rate (recall) against the false positive rate (1 - specificity). AUC is given by the following equation:

$$\text{AUC} = \int_0^1 \text{TPR}\, d(\text{FPR}) \tag{3}$$

or in discrete form,

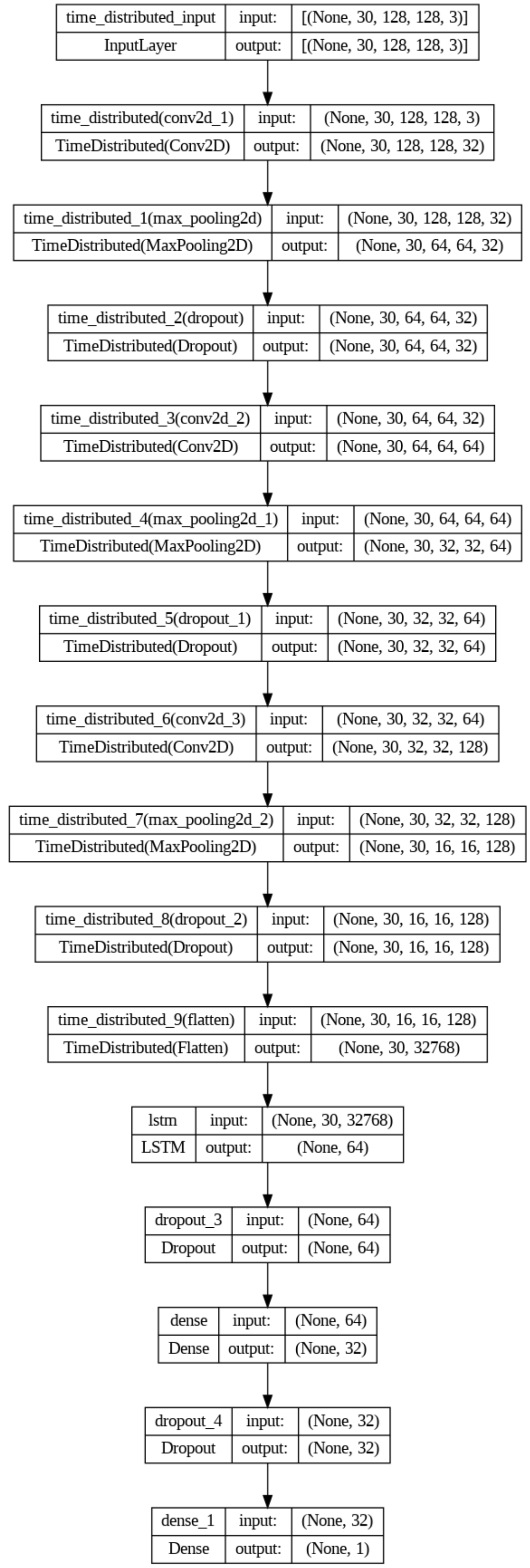

Fig. 1. Model Summary

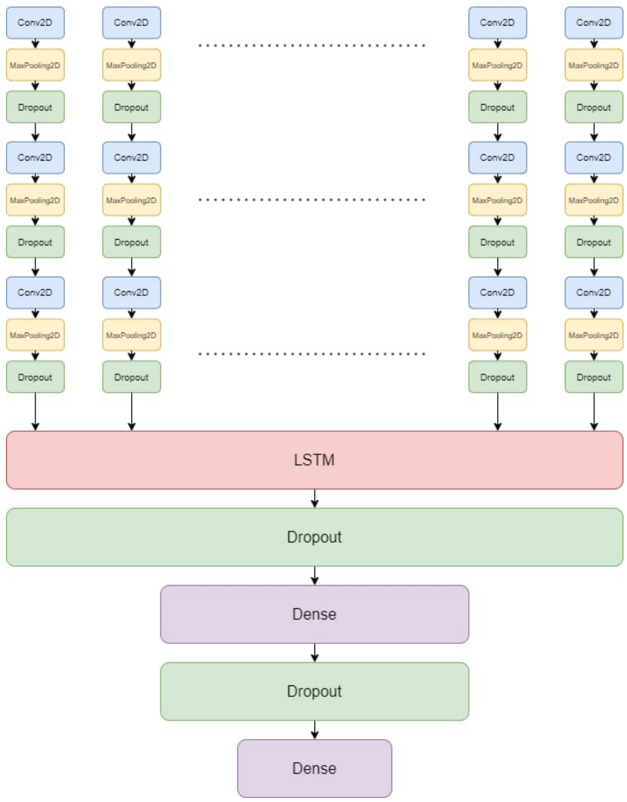

Fig. 2. Model Architecture

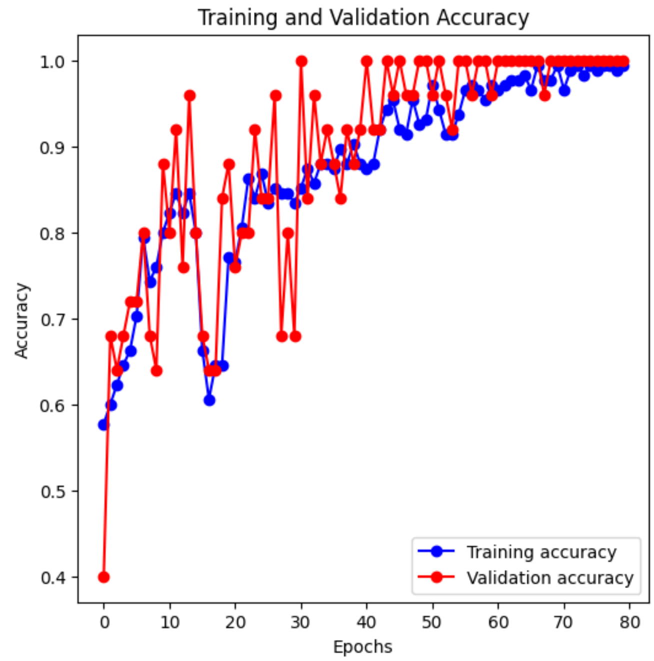

Fig. 3. Training and Validation Accuracy Plot

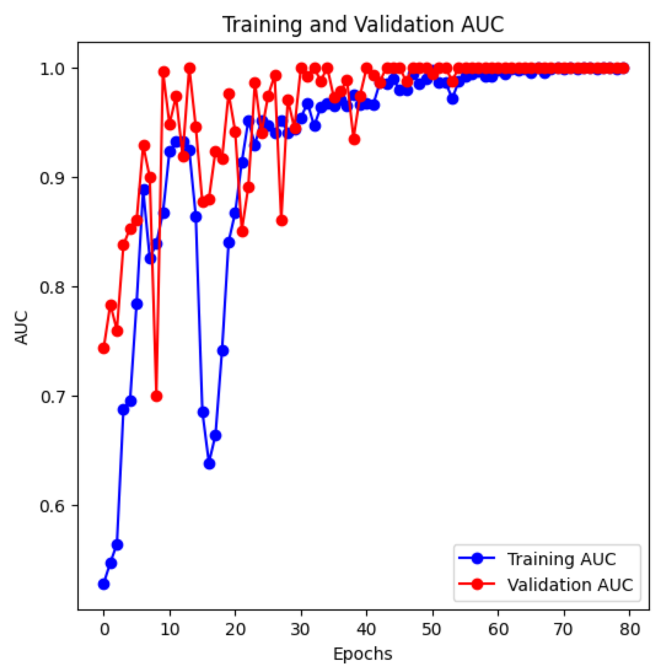

Fig. 4. Training and Validation AUC Plot

$$\text{AUC} = \sum_{i=1}^{n-1} \frac{1}{2}(FPR_{i+1} - FPR_i)(TPR_{i+1} + TPR_i) \quad (4)$$

where TPR (True Positive Rate) is $\frac{TP}{TP+FN}$, FPR (False Positive Rate) is $\frac{FP}{FP+TN}$, and (FPR_i, TPR_i) are the coordinates of the ROC curve points.

Precision measures the proportion of correctly predicted falls among all predicted falls, shown by this equation:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (5)$$

Recall measures the proportion of correctly predicted falls among all actual falls, given by the equation:

$$\text{Recall} = \frac{TP}{TP + FN} \quad (6)$$

In Equations (2), (5), and (6), TP, TN, FP, and FN represent true positives, true negatives, false positives, and false negatives, respectively. These metrics collectively assess the model's performance in fall detection, with accuracy providing a broad view of overall correctness, AUC indicating the model's discriminative ability, and precision and recall offering insights into specific aspects of the model's predictive performance.

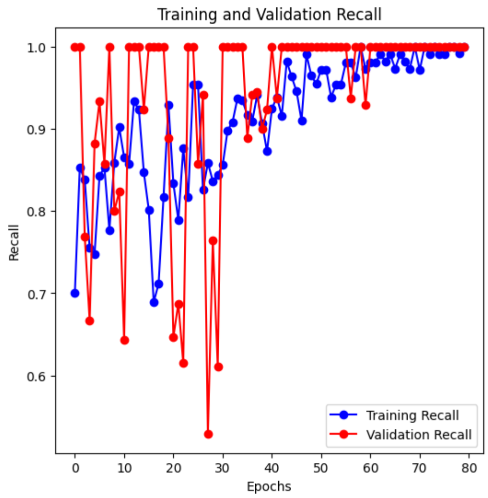

Fig. 5. Training and Validation Recall Plot

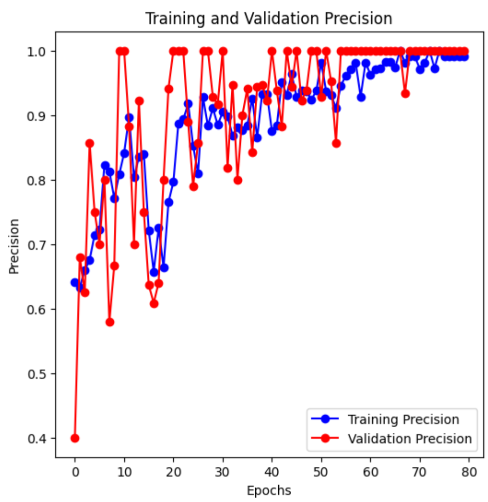

Fig. 6. Training and Validation Precision Plot

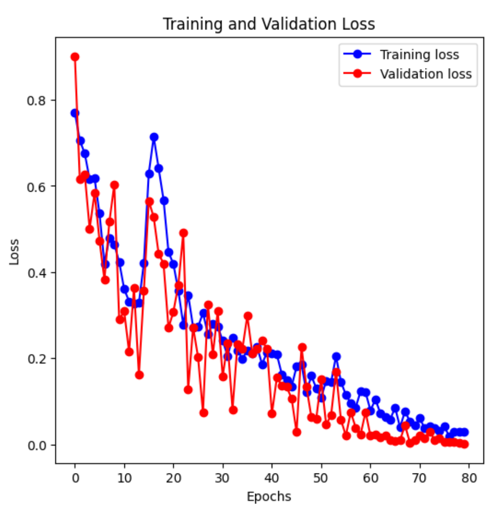

Fig. 7. Training and Validation Loss Plot

LapseNet achieved a training accuracy of 99.43% and a validation and test accuracy of 100% each. The AUC, precision, and recall were 100% in training, validation, and testing. As shown in Fig. 3-6, accuracy, AUC, precision, and recall improved over the 80 epochs that the model was trained on. Moreover, as depicted in Fig. 7, loss continually decreased over the 80 epochs, indicating that the model consistently improved its performance over the training period.

While the model trained on four diverse datasets, it continued to maintain the high level of performance that has been found in other studies using vision-based approaches, such as the 3D-CNN and LSTM approach by Lu et al. [15], which used only two fall datasets, the UR Fall Dataset and Multiple Cameras Fall Dataset, to train their model. A major concern in these studies is a high rate of Type I error (false positives), which can lead to wrongly reporting information to authorities or caregivers, potentially causing unnecessary alarm or intervention. Our model achieved a false positive rate of 0% in training, validation, and testing, demonstrated by its high precision and AUC scores. These metrics indicate that our model minimizes the likelihood of incorrectly predicting a fall when no fall has occurred, thus mitigating the risk of reporting false information.

A key novelty of LapseNet lies in its high accuracy, precision, recall, and AUC. For instance, the 3D-CNN and LSTM architecture tested by Su et al. [16] achieved a 98.06% accuracy on the UR Fall detection dataset but a 94.84% accuracy on the Multiple Cameras Fall Dataset. On the other hand, LapseNet is diversely trained with four datasets and had a higher accuracy of 99.43%, highlighting its generalizability compared to existing solutions. Another notable innovation of LapseNet is its lightweight architecture, which ensures efficient processing and low computational requirements without compromising on accuracy. Chhetri et al. [17] applied enhanced optical flow and pre-trained models to achieve a 40 to 50 ms improvement in processing speed, but only achieved 95.11, 92.91, and 91.1 percent accuracy on the UR Fall Dataset, Multiple Cameras Fall Dataset, and Fall Detection Datasets, respectively. While Maitre et al. [9] developed a 90% accurate fall detection system, its UWB radars are more sophisticated than the standard digital image sensor used in LapseNet because they generate large amounts of data that require significant processing.

V. Conclusion

A. Summary

In this study, we developed a lightweight system to detect collapses in indoor rooms through a vision-based CNN and LSTM architecture. By training with four diverse datasets featuring both fall and ADL scenarios in multiple settings, the results indicate high performance, with validation and testing accuracies of 100%. This performance indicates its potential to significantly improve the safety and wellbeing of the elderly, especially in unsupervised settings. LapseNet's ability to distinguish between ADLs and falls with 0% false positive rate demonstrates its reliability and practicality in real-world scenarios.

B. Limitations

Despite promising results, LapseNet undoubtedly has limitations and a scope for improvement. Although we used four publicly available datasets to make the model robust, the total number of videos may have been insufficient to fully capture the variability in fall scenarios and activities of daily living. Hence, to better train the model, we could have used more participants and videos or simulated falls in different locations and angles. Furthermore, it is possible that our model may have overfit to the data during training, indicated by its 100% accuracy, precision, recall, and AUC across all datasets. Another limitation in the study is that the model was only trained on falls in indoor settings, so it may not perform as well in outdoor or variable environments.

C. Future Work

In the future, we could incorporate acoustic detection or wearable devices in a multimodal approach to increase the system's robustness and accuracy through an additional layer of verification and data fusion. If one modality provides ambiguous results, the other can help confirm the occurrence of a fall. In addition, we could develop a model to identify the specific frames in a video during which the fall occurs in the future. Additionally, we could make a web application that allows users to apply LapseNet to a network of cameras similar to traditional closed-circuit television (CCTV) cameras.

Going forward, we plan to conduct further studies to incorporate these improvements in our model. Altogether, these steps will ultimately contribute to a highly effective tool that can prevent millions of injuries annually worldwide. The successful implementation of LapseNet could lead to a crucial layer of protection in assisted living facilities, private homes, and hospitals.

REFERENCES

[1] Centers for Disease Control and Prevention. Older Adult Falls Data. Older Adult Fall Prevention. Published May 9, 2024.

[2] Vaishya R, Vaish A. Falls in older adults are serious. Indian J Orthop. 2020;54(1):69-74. Published 2020.

[3] Chander H, Burch RF, Talegaonkar P, Saucier D, Luczak T, Ball JE, Turner A, Kodithuwakku Arachchige SNK, Carroll W, Smith BK, et al. Wearable Stretch Sensors for Human Movement Monitoring and Fall Detection in Ergonomics. International Journal of Environmental Research and Public Health. 2020; 17(10):3554. Published May 19, 2020.

[4] Hussain F, Sheng QZ, Zhang W, et al. Activity-aware fall detection and recognition based on wearable sensors. IEEE Sens J. 2019;19(12):1. Published February 2019.

[5] Kaur P, Mishra S, Patel P, et al. Fall detection from audios with audio transformers. Smart Health. 2022;26:100340. Published August 23, 2022.

[6] Khan S, Ahmad A, Muhammad K, et al. An unsupervised acoustic fall detection system using source separation for sound interference suppression. Signal Process. 2014;110(C):199-210. Published August 2014.

[7] Hwang JK, Kang HK, Cho YS, et al. Development of novel algorithm and real-time monitoring ambulatory system using Bluetooth module for fall detection in the elderly. In: Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society. 2005;26(3). Published 2004.

[8] Butt FS, La Blunda L, Wagner MF, Schäfer J, Medina-Bulo I, Gómez-Ullate D. "Fall Detection from Electrocardiogram (ECG) Signals and Classification by Deep Transfer Learning." Information. 2021; 12(2):63. Published 2021.

[9] Maitre J, Bouchard K. Fall detection with UWB radars and CNN-LSTM architecture. J Biomed Health Inform. 2006;25(4):1-11. Published October 2020.

[10] Nuñez-Marcos A, Azkune G, Arganda-Carreras I, et al. Vision-based fall detection with convolutional neural networks. Wirel Commun Mob Comput. 2017;1-16. Published December 06, 2017.

[11] Kwolek B, Kepski M. Human fall detection on embedded platform using depth maps and wireless accelerometer. Comput Methods Programs Biomed. 2014;117(3):489-501. Published December 2014.

[12] Auvinet E, Rougier C, Meunier J, St-Arnaud A, Rousseau J. Multiple cameras fall dataset. Technical report 1350. DIRO - Université de Montréal. 2011. Published January 2011.

[13] Guerrero JCE, España EM, Añasco MM, Lopera JEP. Dataset for human fall recognition in an uncontrolled environment. Data Brief. 2022;45:108610. Published September 17, 2022.

[14] Charfi I, Miteran J, Dubois J, Atri M, Tourki R, Optimized spatio-temporal descriptors for real-time fall detection: comparison of support vector machine and adaboost-based classification. Journal of Electronic Imaging. 22(4);2013. Published 2013.

[15] Lu N, Wu D, Wang C, et al. Deep learning for fall detection: three-dimensional CNN combined with LSTM on video kinematic data. IEEE J Biomed Health Inform. 2019;23(1):314-323. Published February 2020.

[16] Su C, Wei J, Lin D, Kong L, Yong Liang Guan. A novel model for fall detection and action recognition combined lightweight 3D-CNN and convolutional LSTM networks. Pattern analysis and applications. 2024;27(1). Published February 2024.

[17] Chhetri S, Alsadoon A, Al-Dala'in T, Prasad PWC, Rashid TA, Maag A. Deep learning for vision-based fall detection system: Enhanced optical dynamic flow. Computational Intelligence. 2020;37(1):578-595. Published March 18, 2021.