Senior Fall Detection Through Vision and Vibration Techniques with Raspberry Pi

Zeyu Liu (student)[1*], Yourui Shao (student)[2], Darren Ng (mentor)[3]

1. Valley Christian High School, 100 Skyway Dr, San Jose, CA, 95111, United States

2. BASIS Independent Silicon Valley, 1290 Parkmoor Ave, San Jose, CA, 95126, United States

3. The University of California Merced, 5200 Lake Rd, Merced, CA, 95343, United States

*bennyzeyuliu@gmail.com

Abstract

The number of seniors living alone without caretakers is on the rise. With falls accounting for a significant percentage of injuries and mortality in elders, fall detection devices are critical, allowing immediate medical attention through promptly alerting caretakers. We reviewed current fall detection methods and improved their accuracy by implementing new considerations. A “double-checking” system minimizes errors by combining vision and vibration-based detection reports using decision trees. Adjustments to existing vision-based detection methods were made by improving camera positioning and introducing angular velocity. We reviewed various vision and vibration-based fall detection models, analyzed their inconsistency patterns, and optimally consolidated their results.

Computer Science; Senior Fall Detection; Neural Network; Long Short-Term Memory; Pose Detection

Introduction

Background

Over 25% of the elderly (65+ years old) fall annually in the United States, resulting in over 27,000 related deaths.[1] This figure is expected to rise, as the global senior (60+ years old) population is projected to double in the next three decades, from 12% in 2015 to 22% in 2050.[2] As such, falling among the elderly is an issue of concern for health officials and caregivers.

A scenario of concern is when a fallen individual living alone cannot get up and call for help. Having a “long lie”—being unable to get up after falling—positively correlates with the statistical likelihood of death within twelve months, from 25% of those who could get up dying to 50% of those who could not. Stemming from the fear of falling with a “long lie,” past falls may have, in addition to potential health effects, possible psychological trauma on individuals, such as avoiding everyday locomotive activities like reaching overhead and bending down.[3,4]Such fear decreases one’s mobile activity and quality of life.[3] Even when able to call for help via manual alarms, seniors are still likely to attempt to rise by themselves (with 80% in one study not using their alarms), placing greater physical stress on their bodies.[5]

Existing Fall Detection Methods

Generally, three types of fall detection devices have been implemented: wearable, camera-based, and ambiance (Table 1).[6] Many seniors have expressed dissatisfaction with the intrusiveness of wearable devices despite their accessibility and low cost.[6] Additionally, removing the wearable device for activities such as showering or sleeping is inevitable and disrupts its effectiveness. The vision-based approach overcomes the latter problem. Cameras placed on ceilings or walls for extensive video coverage continuously monitor individuals for potential falls. Lastly, the ambiance approach utilizes multiple sensors attached to walls or the floor and detects environmental changes such as floor vibrations, of which high magnitudes indicate a fall.

Vibration-based detection is limited in its detection range and the ability to identify different falling objects.[7] Heaving objects falling and people performing

Features	Wearable	Vision	Ambience
Cost	Low	Low to medium	Medium to high
Intrusive	Yes	Possibly	No
Live	✓	✓	–
Setup	Easy	Easy	Complex
Visuals	×	×	✓

Table 1: Comparing cost, intrusiveness, rapidness, ease, and visibility of types of fall detection devices.[6]

intensive physical activity may be misidentified as falls. Falls dampened by objects may also be undetected. With vision-based detection, the camera's field of view may be obstructed by large furniture and sectioned living spaces require multiple cameras for full coverage. As such, there exist shortcomings of each method that could be improved on.

Vibration-based

Introduced by Alwan *et al.* in 2006, the vibration-based fall detector was the first passive, non-intrusive method of fall detection.[8] Using a piezoelectric sensor: the study confirmed that the vibrations produced by a fall, simulated by dropping a dummy, are distinguishably different from those of common motions and other falling objects in vibration features such as frequency, amplitude, duration, and succession. Alwan *et al.* proposed that users could toggle software options for different floor materials to adjust the predicted fall pattern so that material-induced vibration discrepancies are accounted for.

Subsequently, Liu *et al.* used a multi-feature semi-supervised support vector machine (MFSS-SVM) classifier for classifying fall and no-fall states using vibration patterns.[7] In this algorithm, unlabeled vibration data is used alongside labeled samples when the quantity of labeled data is insufficient for the classifier. The few labeled data are used to train the SVM classification model, through which unlabeled data is labeled with varying confidence. Unlabeled data with higher certainty are then used as "pseudo-labeled" data in conjunction with labeled data in the classifier model. Using the features maximum peak value (A_{max}), "energy" ($\int |a(t)|dt$, the area under the curve of accelerometer data), and the sensor correlation coefficient of two accelerometers, the model was able to identify falls in both laboratory tests and a benchmark test with simulated measurements.[7]

Vision-Based

CNN Technique One vision-based fall detection model is the Convolutional Neural Network (CNN) technique.[9] Artificial intelligence (AI) neural networks mimic the human brain by learning through trial and error. A CNN model is a type of neural network that primarily classifies images. With fall detection, the CNN technique is distinct from other models in its ability to measure fall duration. The CNN model, given a video, separates the individual from the background, splices the video into frame sequences, and categorizes the individual in the video splits as in one of four states: standing, falling, fallen, and not moving. The video is classified as a fall if these states occur sequentially in the video splits.

LSTM Technique The Long Short Term Memory (LSTM) technique involves observing human pose and body geometry to detect falls.[10] A LSTM model is used as they are suitable for learning from long-term dependencies between time steps of data. Two vectors are drawn on the individual in each video frame through manual annotations or an image detection model. One vector extends from the hip to the neck while the other vector is parallel to the ground and originates from the hip. The LSTM model observes the angle between these two vectors and detects a fall when the angle becomes unnatural.

SVM Technique The Support Vector Machine (SVM) technique is very similar to the LSTM technique.[11] While LSTM uses a neural network to evaluate the angle between two vectors, which updates its long-term memory based on the calculations of the weights and biases of its short-term memory, SVM works by identifying divisions between clusters of data to classify it into different categories. This method is also used in vibration-based fall detection.

Depth-based In the depth-based method (Ma et al.), the human silhouette is extracted from an image and represented with a curvature scale space (CSS)-based representation. Next, the human silhouette is subjected to a set of Gaussian functions to obtain a set of convolved curves. The extreme learning machine (ELM) model then uses the convolved silhouettes to determine if a fall has occurred. This strategy performs comparably to the SVM model and allows for differentiation between fallen, sitting down, and lying down states.

Methods

The accuracies of vision-based models could be increased with certain adjustments. Additionally, vision-based and vibration-based fall detection methods have different scenarios in which they specialize in accurate detection. The union of the two methods' high-accuracy scenarios covers a more extensive set of scenarios than each method does separately. Thus, in addition to changes to the vision-based detection method, using the two methods simultaneously could yield better detection accuracy overall.

Improving Vision-Based Detection (VIS)

The method of Beddiar *et al.* is similar to existing solutions.[11] The model considers the angle (α) between two vectors—one from the hip to the neck ($\vec{U}$) and one from the hip directed parallel to the ground ($\vec{V}$)—calculated with the Law of Cosines (Equation 1). When sitting or standing up, α nears $\pi/2$ radians. However, when one is falling, α will begin approaching either 0 or π radians depending on the person's

location relative to the camera.

$$\alpha = \arccos\left(\frac{u \cdot v}{\|u\|\|v\|}\right) \quad (1)$$

However, the model was unable to differentiate precisely between the actions of falling and lying down voluntarily, which differ in the duration of those movements.[11]

Falling Speed In one study, video frames are translated into a xy-plane, and the individual's falling speed is calculated by the motion of their neck relative to their torso.[10] Possible falls are indicated by irregular movement when the velocity of the neck, calculated using coordinates from the xy-plane, surpasses a set threshold.

Tracking the motion of a person's neck is not optimal, as a 3-dimensional environment is compressed into a 2-dimensional video. Therefore, angular velocity between the mentioned $\vec{U}$ and $\vec{V}$ vectors should be calculated instead. Angular velocity ω, calculated using α from $\vec{U}$ and $\vec{V}$: $\omega = \frac{\Delta\alpha}{\Delta t}$, would replace linear neck velocity. If ω exceeds a set value, the angle changes irregularly, indicating a plausible fall.

Camera Position Unlike falling in the horizontal plane of the camera's field of view, falling in or away from the camera's direction would affect α less and may go undetected. Objects in the background may be misdetected as an individual such that the model loses track of the person while or after falling. Thus, a higher camera placement should be used as it increases $\Delta\alpha$ observed regardless of fall direction.

Combining Technologies

Both vision-based and vibration-based approaches have their distinct limitations. Vision-based detection is limited as the cameras' field of view may be blocked. For ambiance devices such as vibration detectors, each sensor's range of detection is constrained and there is a degree of inconsistency. The two detection methods—vibration-based (VIB) and vision-based (VIS)—could be combined to overcome these limitations and increase accuracy. However, it must be addressed that the reports from the two methods may differ within the same timeframe. Using metrics of the approaches of Liu *et al.* and Beddiar *et al.* for VIB and VIS methods, respectively, the course of action for contradicting reports could be determined.[7,11]

Vision-Based The Angle + Distance + LSTM method is used to compare the VIB model. With a recall (detected falls out of all falls—T/T) of 89%, we can determine that T/F (false negative—falls occurred but were unreported) is 11%.[11] There is an accuracy (percentage of correct identifications) of 84.6%, which gives us the following equation:

$$84.6\% = \frac{\text{T/T} + \text{F/F}}{\text{T/T} + \text{T/F} + \text{F/T} + \text{F/F}} = \frac{\text{T/T} + \text{F/F}}{200\%}. \quad (2)$$

The sums T/T + T/F and F/T + F/F are normalized as 100%. With T/T known (89%), F/F $= 84.6\% \cdot 200\% - 89\% = 80.2\%$, and thus F/T (false positive—no falls occurred but were reported) is 19.8%.

Vibration-Based As only the training execution time of this model increased significantly when the labeling rate increased (from 20% to 80%). At the same time, accuracy increased significantly: we will consider the metrics of an 80% labeling rate model.[7] In this case, the misreporting rate—F/T—is 7.4%, while the missing report rate—T/F—is 0.8%. Thus, we can deduce that T/T and F/F (correctly no falls reported) are 99.2% and 92.6%, respectively.

For each method, the metrics in Table 2 were obtained using different tests, with VIS using the Le2i dataset and VIB using various drop-tests and other active motions. Nevertheless, Table 2 illustrates the shortcomings of each detection method.

Case 2, Table 3 The VIB sensor tends to misreport falls, particularly with falling objects.[7] Thus, a fall is asserted only when the vibration amplitude exceeds an assigned A_H value that falling objects are unlikely to cause, and the VIS model is not highly confident there was no fall.

Case 3, Table 3 It is much more likely that the VIS model misreports a fall (19.8%) than the VIB model fails to detect one (0.8%). The reliability of the VIB model could fall short of that of the VIS model when the amplitude of vibration, A, is dampened (e.g. by falling on a cushioning object). As such, during the time frame in which the camera detects a fall, the amplitude below the typical threshold but reasonable in this circumstance ($A \geq A_L$) would nevertheless report a fall.

		Predicted			
		VIB		VIS	
		T	F	T	F
Actual	T	.992	.008	.89	.11
	F	.074	.926	.198	.802

Table 2: Adjusted confusion matrix of fall (T) and no-fall (F) states for existing VIB and VIS detection methods. Per the model, each row sums to 1.

	Case	Result
1	VIB: T VIS: T	T always
2	VIB: T VIS: F	$\begin{cases} \text{T} & \text{if } A \geq A_H \text{ and } C_T^{\text{VIS}} \geq 10\%, \\ \text{F} & \text{otherwise.} \end{cases}$
3	VIB: F VIS: T	$\begin{cases} \text{T} & \text{if } A \geq A_L, \\ \text{F} & \text{if } A < A_L. \end{cases}$
4	VIB: F VIS: F	F always

Table 3: Course of action for report combinations of VIB and VIS models. A_H has increased amplitude threshold, A_L as decreased amplitude threshold, and C_T^{VIS} as VIS fall confidence level.

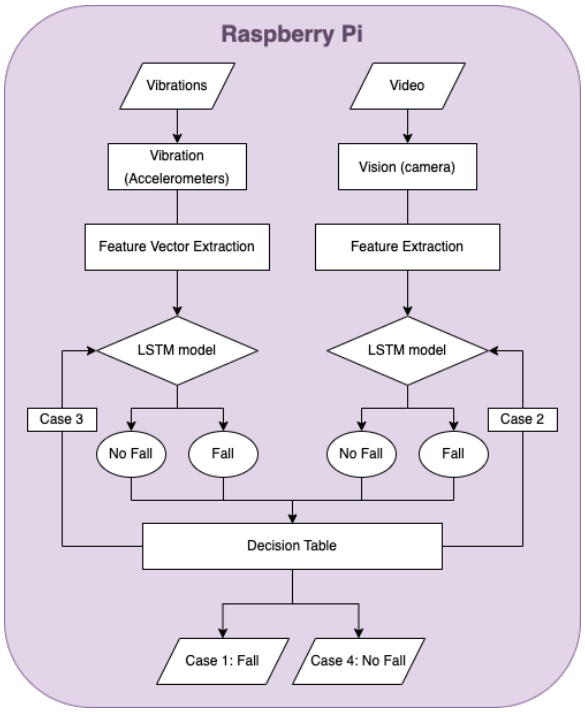

Figure 3: The architecture for a combined fall detection system. Two information channels—vibration and video—feed into their respective models. "Decision Table" refers to Table 3.

Implementation

Hardware The model combinations and fall alerting channels should be placed on a standard computing device, such as the Raspberry Pi. As illustrated in Figure 3, vibration and vision sensors are located on the Raspberry Pi. Captured data is then sent to the respective models to be categorized as a fall or no-fall. As in Table 3, if both the VIS and VIB models report a fall, then a fall is alerted. If model outputs differ, thresholds are adjusted to retest the data, increasing the likelihood of an accurate final report.

VIS Model The velocity function is replaced by an angular velocity function as described in Falling Speed.[10]

VIB Model A VIB LSTM model was trained. The data collected for preparatory training—a chart of vibration patterns (Figure 2)—was collected separately from that used with the VIS model. 18 vibration data points were collected per second, with the average of the 18 points being the vibration score for that second. LSTM input data consisted of labeled five-second intervals of vibration scores.

Data Collection

VIS and VIB data were collected separately initially. Featuring object obstruction and visual noise, VIS data varied and were more realistic to common falling scenarios than existing datasets (Figure 1) One hundred fifty frame videos at 25 frames per second were used to estimate two seconds per fall with four additional buffer seconds. 120 and 90 framed videos resulted in lower accuracy scores and varied results. Specific cases pertaining to the weaknesses of the VIS model were additionally recorded. Fall data varied in fall direction and fall speed and no-fall data included "fake falls" in which a person would slowly sit down on the mattress to test the effectiveness of using ω as a feature.

In total, 241 video segments and eight hours of vibration data were used to train their respective models. Then, VIS and VIB data were recorded simultaneously for testing in Table 3. Initially, 29 data points were used for testing but were expanded to 85 for more accurate results.

Results and Discussion

Performance Matrices

Accuracy, precision, recall, specificity, and F_score are key performance metrics that determine the effectiveness of the detection methods.

Accuracy: Percentage of correctly detected falls/no-falls over the total data.

$$\text{Accuracy} = \frac{\text{True Positive + True Negative}}{\text{Total Number of Data}} \tag{3}$$

Precision: Percentage of correctly detected out of

falls detected.

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive + False Positive}} \quad (4)$$

Recall: Percentage of falls correctly detected.

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

Specificity: Percentage of no-falls correctly detected.

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

F_score: A measurement of accuracy, the harmonic mean of precision and recall.

$$\text{F\_score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision + Recall}} \quad (7)$$

Experimental Results

Vision-Based Greater background noise and no-fall data in which the subject laid down to test the added ω feature resulted in lower precision and recall than tests in other studies (Tables 4 and 5).

The ω + LSTM VIS model produced an accuracy of 80% when n=85. Precision would increase when the model can fine-tune the difference in ω between falling and laying down with more data. Background noise also contributed to false negatives (Figure 4) when the model failed to keep track of the individual's $\vec{U}$ and $\vec{V}$ vectors due to color similarities to the background.

Vibration-Based Because all fall data was collected on a mattress, the vibration score threshold (700) is lower than what is observed in an actual fall scenario. The specificity of the vibration-based model (97.7%, n=85) was higher than the accuracy of the vision-based model (86.0%, n=85), with only one false positive, whereas VIS produced 6, as seen in Figure 4.

Both detection methods produced low recall rates (73.8-76.2%, n=85) and missed a significant number of falls.

Combined Results

Combining the two systems generally improved detection metrics (Tables 4 and 5). For both n=29 and n=85, accuracy and recall enhanced as the number of false negatives declined due to the "double-checking" system. The combined method's overall accuracy was 91.8%, with a recall score of 88.1%. However, precision and specificity declined when the two methods were combined with an increase in the number of false positives, compared to using only one method. As VIB tends to have low false positives, an error likely occurred in Case 3 (Table 3) when VIS misdetected a fall and VIB passed the decreased A_L threshold.

Discussion

Overall, combining the VIS and VIB detection systems resulted in improved metrics, which would vary with more extensive data sets. Though the results could be better than that of other studies due to the limited training data while increasing problem complexity, it demonstrates the effectiveness of combining multiple detection methods. The decision tree (Table 3) covers each model's inaccuracies using adjusted thresholds, which should be modified to lower the number of false positives. False positives alert to a fall when there is none, which is resource-costly for caretakers and emergency services. False negatives pose a more significant concern when an individual falls without detection.

It is thus crucial to improve the recall rate, reducing false negatives, for both VIS and VIB models to yield overall improvement. Larger samples of varied training data and optimization in object identification to ignore background noise are needed. Furthermore,

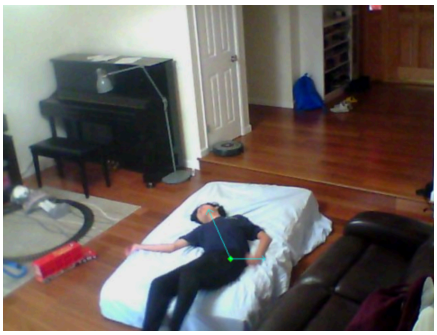

Figure 1: Example of a captured frame of a fall.

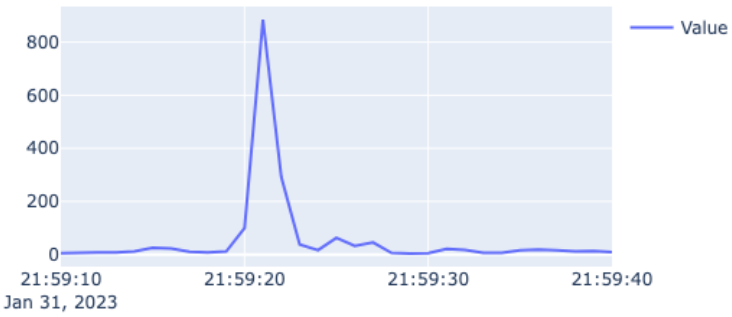

Figure 2: Example of a vibration anomaly (fall).

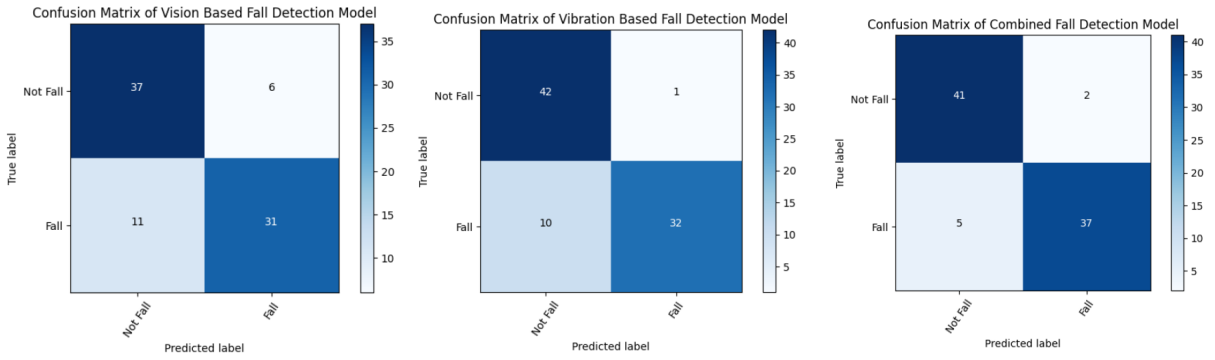

Figure 4: VIS model, VIB model, and Table 3 combined model confusion matrix.

Model	Acc.	Prec.	Recall	Spec.	F_score
Vision-based LSTM	.828	.813	.867	.786	.799
Vibration-based LSTM	.793	.909	.667	.929	.769
Combined	**.931**	**.933**	**.933**	.929	**.933**

Model	Acc.	Prec.	Recall	Spec.	F_score
Vision-based LSTM	.800	.838	.738	.860	.785
Vibration-based LSTM	.871	.970	.762	.977	.853
Combined	**.918**	.949	**.881**	.953	**.914**

Table 4: Performance result comparison for the VIS and VIB LSTM models from size 29 test data (n=29).

Table 5: Performance result comparison for the VIS and VIB LSTM models from test data with n=29. The green text highlights areas of gain from Table 4.

auto-adjusted thresholds for VIB detection and the decision table (Table 3) would increase overall accuracy.

Conclusion

Introducing a “double-checking” combination system using case-action algorithms improved fall detection accuracy and F-scores overall with both test sizes. In addition, this “double-checking” system could be helpful in other fields valuing accuracy.

Acknowledgments

We are incredibly grateful to our parents, who provided us the opportunities and resources to complete this project. Special thanks to Darren Ng, a master’s student at the University of California Merced, who pre-viewed our paper and gave helpful feedback on writing and organization.

References

(1) Bergen, G.; Stevens, M. R.; Burns, E. R. *Morbidity and Mortality Weekly Report* **2016**, *65*, 993–998.
(2) Kannus, P.; Parkkari, J.; Niemi, S.; Palvanen, M. *American Journal of Public Health* **2005**, *95*, 422–424.
(3) Lachman, M. E.; Howland, J.; Tennstedt, S.; Jette, A.; Assmann, S.; Peterson, E. W. *The Journals of Gerontology Series B: Psychological Sciences and Social Sciences* **1998**, *53*, P43–P50.
(4) Williams, V.; Victor, C. R.; McCrindle, R. *Current gerontology and geriatrics research* **2013**, *2013*.
(5) Fleming, J.; Brayne, C. *Bmj* **2008**, *337*.
(6) Tanwar, R.; Nandal, N.; Zamani, M.; Manaf, A. A. In *Healthcare*, 2022; Vol. 10, p 172.
(7) Liu, C.; Jiang, Z.; Su, X.; Benzoni, S.; Maxwell, A. *Sensors* **2019**, *19*, 3720.

(8) Alwan, M.; Rajendran, P. J.; Kell, S.; Mack, D.; Dalal, S.; Wolfe, M.; Felder, R. In *2006 2nd International Conference on Information & Communication Technologies*, 2006; Vol. 1, pp 1003–1007.
(9) Alam, E.; Sufian, A.; Dutta, P.; Leo, M. *Computers in Biology and Medicine* **2022**, *146*, 105626.
(10) Wangwiwattana, C. *International Journal of Development Administration Research* **2019**, *2*, 12–22.
(11) Beddiar, D. R.; Oussalah, M.; Nini, B. *Journal of Visual Communication and Image Representation* **2022**, *82*, 103407.

Authors

Zeyu Liu is a high school junior who enjoys research and computer science. He is part of his school's International Space Station Research Lab and likes reading in his free time. He is taking Multi-Variable Calculus and AP Physics C and finds challenging problems intriguing.

Yourui Shao is a high school sophomore who enjoys problem-solving, mathematics, and making art. She is also an avid sleeping enthusiast and loves proofreading her own overly-long sentences.