Evaluation and development of a low power fall detection algorithm implemented in a hearing implant

Fabian Schießendobler

***Abstract*—As the world's population ages, age-related health risks are increasing, particularly the risk of falls among older people. Falls pose a significant threat to the health and well-being of older adults and require prompt support for prevention and mitigation. Taking advantage of the widespread use of hearing aids and implants among older adults, this paper proposes the development of an effective fall detection algorithm tailored for integration into hearing implant systems. Machine learning algorithms, particularly those based on vision and sensors, have shown promise in fall detection systems. However, integrating such systems into hearing implant systems presents several challenges, including accurately detecting falls with head-mounted sensors and implementing power-saving algorithms. This innovative algorithm addresses these challenges by accurately detecting falls, near falls, and activities of daily living (ADL) while minimising energy consumption, thereby providing potentially life-saving assistance. Using a dataset from Inertial Measurement Units (IMU) attached to participants' heads, activities were categorised into falls, near falls and ADL. Different feature subgroups were investigated, and a hybrid Convolutional Neural Network-Long Short-Term Memory (CNN-LSTM) architecture was developed for classification. Hyperparameter tuning of the machine learning algorithms and downsizing for low power consumption were performed, resulting in the development of an efficient fall detection algorithm suitable for hearing implant systems. The proposed algorithm, which uses acceleration data with magnitude, demonstrates superior classification accuracy performance while maintaining power consumption efficiency. The CNN-LSTM architecture effectively captures spatial and temporal patterns, resulting in high accuracy in classifying falls and ADL. These findings contribute to the advancement of fall detection systems for hearing implant systems, ultimately improving older people's safety and quality of life. Further research is recommended to improve the classification of near falls and validate the algorithm's effectiveness in real-world settings so that it can be implemented in a market-ready device.**

***Index Terms*—Fall Detection, Hearing Implant, Machine Learning, Supervised Learning, Timeseries Data.**

I. Introduction

As healthcare and living conditions improve, the global average life expectancy is increasing, leading to a shift in the demographic composition of societies towards an older population. By 2050, Europe will likely have 164 million individuals aged 65 and above [1]. Furthermore, the global population of those aged 60 and over is expected to be higher than those younger than 15 years [2]. With this demographic change comes an increased prevalence of age-related health risks, particularly the risk of falling among elderly individuals [3]. Falls pose a significant risk to the health and well-being of older adults, often resulting in injuries, decreased mobility, and a decline in overall quality of life [4]. They are the primary cause of injury-related fatalities among seniors aged 79 and above and the second most common cause of unintentional injury-related deaths across all age groups [5]. Moreover, the absence of prompt assistance during a fall increases the severity of injuries and extends recovery for older adults [6].

Currently, the use of hearing aids and implants among older adults has become more common to address age-related hearing impairments [7]. While these devices enhance auditory perception, they also present an opportunity to integrate fall detection capabilities, leveraging the technology already worn by many seniors. Therefore, there is a need for an effective fall detection algorithm explicitly designed for hearing implant systems. Such an algorithm would need to accurately identify falls, near falls and activities of daily living (ADL) while minimising energy consumption to provide potentially life-saving assistance. Addressing these challenges will improve the safety and well-being of elderly individuals, empowering them to maintain independence and autonomy in their daily lives. In recent years, machine learning algorithms have played a significant role in advancing technologies for fall detection. There are two primary categories of fall detection based on machine learning: Vision-based fall detection and sensor-based fall detection [8].

Vision-based fall detection systems use cameras to monitor and analyse human activities, detecting falls by identifying specific motion patterns or anomalies in the video feed. These systems use computer vision techniques and machine learning algorithms to distinguish falls from everyday activities. Vision-based fall detection is more accurate at spotting falls because it captures detailed environmental information but comes with higher infrastructure costs. Sensor or wearable fall detection systems use accelerometers, gyroscopes, pressure sensors and magnetometers to monitor an individual's movements. The system analyses the data collected from these sensors to detect sudden changes or anomalies that may indicate a fall. Machine learning algorithms distinguish between everyday activities and fall events, offering a non-disturbing and effective approach. Embedding sensor-based fall detection into wearable devices is seamless and cost-effective due to the use of a single sensor unit. By continuously monitoring the movements, fall detection systems provide timely alerts to caregivers or emergency services. Furthermore, research

F. Schießendobler is with the Department of Medical and Health Technologies, MCI, Innsbruck, Austria, e-mail: sf3254@mci4me.at.

proposed approaches for a combination of wearable devices and machine vision [9], where vision-based solutions are used to validate the predictions of the sensor-based approach [10]. Using machine learning for fall detection shows promising potential in improving functional outcomes and daily activities [11], [12]. This technology identifies falls and contributes to proactive measures for preventing injuries, thereby significantly enhancing the overall quality of life for individuals at risk of falling.

Recent studies have proposed various methods for fall detection using different technologies. For instance, Wang and Jia proposed a video-based fall detection method by using the YOLOv3 network with a pre-labelled dataset [13]. Lee and Mihailidis focused on indoor fall detection, using tracking techniques with connected-components labelling to extract relevant features from silhouette data [14]. Wang et al. presented a Wi-Fi-based system for fall detection that uses digital signal processing and a Support Vector Machine (SVM) algorithm to analyse the Channel State Information (CSI) to identify abnormal sequences indicating falls [15]. Kwolek and Kepski utilised a triaxial accelerometer alongside depth maps from a Kinect camera, employing an SVM classifier to process data from both sources [16]. Furthermore, Vallejo et al. developed a fall detection method based on Artificial Neural Networks (ANN) utilising data from a 3-axis accelerometer [17]. Sengto and Leauhatong used a similar accelerometer setup, feeding a backpropagation Multilayer Perceptron (MLP) to distinguish between falls and daily-life activities of older adults [18]. The accelerometer was attached to the waist of the study participants. In another approach, Li et al. proposed a Deep Learning (DL) model, combining a Temporal Convolutional Network (TCN) and Gated Recurrent Unit (TCN-GRU) architecture, trained on datasets such as MobiAct and Mosi-F for feature extraction and classification [19]. Most sensor-based fall detection algorithms focus on sensors positioned at the chest, waist or wrist level. Despite ongoing research in developing fall detection systems for hearing aids, limited progress has been made due to challenges with accurately detecting falls when sensors are positioned on the head and difficulties in implementing energy-saving algorithms. In a study conducted by Burwinkel et al., the efficacy of fall detection using inertial sensors in hearing instruments was evaluated by comparing them against traditional fall detection pendants [20]. The results suggest that hearing instruments with embedded fall detection technology could effectively replace more traditional and visible devices, providing a discreet and accurate method for fall detection and potentially reducing the incidence of long lies after a fall among older adults. A notable example is Starkey's Livio Edge AI hearing aid, which integrates a fall detection algorithm using an Inertial Measurement Unit (IMU) and virtual sensor streams to differentiate between falls and daily activities [21].

Burwinkel et al.'s study has found promising results, especially for implementation in a hearing implant [20]. To the best of our current knowledge, no research has been done to develop a low-power fall detection algorithm embedded in a hearing implant system capable of differentiating between falls, near falls and ADL. This paper presents a novel approach to effectively detect and prevent falls among individuals, marking a significant stride towards enhancing safety and quality of life for those with hearing impairments.

II. Methods

A. Data Collection

Özdemir and Barshan collected the dataset used in this study [22]. The collected dataset is available under the Creative Commons Attribution 4.0 International (CC BY 4.0) license, promoting accessibility and collaborative exploration. Seventeen volunteers (10 male, seven female), with an average age of 21.94 $\pm$ 1.98 years, an average height of 171.64 $\pm$ $7.82\,\mathrm{cm}$ and an average weight of 65 $\pm$ $13.85\,\mathrm{kg}$, contributed to the dataset, representing a young age group with diverse body types. Each participant wore a sensor (Xsens MTw, Movella) securely attached to their head, which housed an IMU capable of capturing 23 measurements. A total of 36 activities were performed, each repeated five times by every participant to ensure the dataset's reliability and robustness. This resulted in 3060 instances, each consisting of $20\,\mathrm{s}$ of IMU data. Participants were instructed to execute the activities naturally, imitating the movements of older people. During the performance of each activity, the sensor continuously recorded IMU data at a sampling rate of $25\,\mathrm{Hz}$ to capture detailed motion dynamics.

B. Data Processing

In this study, the initial 36 activities were reduced and categorised into 13 falls, 13 ADL and three near falls, resulting in 2286 instances, see Figure 1.

The IMU data was preprocessed by selectively extracting significant timeseries and sensor signals such as triaxial accelerometer and gyroscope data. A sliding window technique was subsequently applied, resulting in 10021 instances. Each instance spanned $1.44\,\mathrm{s}$, representing the average duration of a single movement as outlined in Figure 1. To achieve uniform distribution, falls had an overlap of $0.64\,\mathrm{s}$, near falls $0.72\,\mathrm{s}$, and ADL $1.28\,\mathrm{s}$. The overlaps were chosen to balance the number of windowed datasets for the different activities. The datasets were shuffled randomly with a seed value of 42 for reproducibility and to reduce the risk of model overfitting before being fed into the machine learning algorithms.

C. Investigation Feature Subgroups

To maintain low power consumption, it is essential to assess whether triaxial gyroscope data is necessary due to its additional power consumption compared to an accelerometer. Therefore, the windowed datasets underwent postprocessing, resulting in four subgroups:

- Timeseries data including triaxial accelerometer values,
- timeseries data including triaxial accelerometer values alongside their magnitude,
- timeseries data including triaxial accelerometer and gyroscope values,
- and timeseries data including triaxial accelerometer and gyroscope values along with their respective magnitudes.

Fall Actions
Vertical falling forward on the floor
Vertical falling forward on the floor with arm protection
Vertical falling on the knees and then lying on the floor
Vertical falling on the floor, ending in right lateral position
Vertical falling on the floor, ending in left lateral position
Vertical back-falling on the floor, ending lying
Vertical back-falling on the floor, ending lying in right lateral position
Vertical back-falling on the floor, ending lying in left lateral position
Vertical right-falling on the floor, ending lying
Vertical left-falling on the floor, ending lying
Vertical standing on a podium going on the floor
Falling on the floor following a vertical trajectory
Slowly slipping down a wall
ADL
Standing
Walking forward
Walking backwards
Running
Squatting, then standing up
Bending, about 90 degrees
Bending to pick up an object on the floor
Walking with a limp
Sit on a chair
Sit on a bed
Lying on a bed from standing
Lying
Rising from bed
Near Falls
Vertical falling on the knees, with recovery
Stumbling, with recovery
Trip-over

Fig. 1. Activities for training the machine learning algorithm.

For comparative analysis across the different feature subgroups, a hybrid Convolutional Neural Network-Long Short-Term Memory (CNN-LSTM) architecture was developed, and its performance was evaluated in terms of accuracy and loss function. The dataset underwent two splits: an 80-20 split, allocating 80% of the timeseries data for training and testing while reserving 20% for validation, and a separate split dividing the dataset into a training and test set with 16 participants and a validation set involving a single participant (referred to as 16-1 split).

D. Hypertuning of Machine Learning Algorithms

The dataset, which consists of timeseries data containing triaxial accelerometer values and magnitude, was further used to evaluate three different machine learning algorithms for sensor-based fall detection to determine the most effective model. These algorithms include:

- A Convolutional Neural Network (CNN),
- a Long Short-Term Memory (LSTM) model,
- and a CNN-LSTM hybrid model.

All models underwent hyperparameter tuning using Keras-Tuner to optimise the performance based on the underlying dataset [23]. The dataset was initially split in an 80-20 ratio. Here, 20% of the data was used for validation, while the remaining 80% facilitated training and testing within the neural network framework. A further refinement followed, in which the 80% segment was divided into a further 80-20 split. This secondary split allocated 80% to training the neural network and reserved 20% for testing its effectiveness. In connection with the RandomSearch tuner, the secondary split is used to find the optimal hyperparameters yielding the lowest validation loss for the dataset. Once the best hyperparameter combination was found, the best epoch based on the validation accuracy was selected to train the model. The effectiveness of the chosen hyperparameters and epoch was then validated using the validation data from the initial 80-20 split and the 16-1 split to determine which model achieved the highest accuracy with the lowest loss function.

E. Low-power Fall Detection Algorithm

To effectively deploy a machine learning algorithm on a Microcontroller Unit (MCU), it is essential to prioritise low power consumption and minimal flash space usage. Deployment on an MCU requires downsizing hypertuned algorithms to reduce the number of layers while maintaining optimal performance. Specifically, the aim was to downsize the fall detection algorithm to less than $110\,\mathrm{kB}$. To achieve this goal, a Teacher-Student-Model framework was implemented to distil the knowledge from the hypertuned teacher model to the downsized student model. Knowledge transfer from the teacher model to the student is achieved by minimising the loss function designed to align the softened teacher logits (unnormalised predictions) and the ground truth labels [24]. The softening of the logits is achieved by introducing a "temperature" scaling factor within the softmax operation, thereby smoothing the probability distribution and revealing the inter-class relationships embedded in the teacher's learning [24]. The performance evaluation of the low-power fall detection algorithms was then validated using the validation data from the initial 80-20 split and the 16-1 split to determine which model achieved the highest accuracy, Precision, Recall and F1-Score with the lowest loss function.

III. Results

A. Feature Set Selection

Investigation of the feature subgroups revealed that the most efficient and effective configuration included timeseries data containing triaxial accelerometer values and their magnitude, see Figure 2. This choice was made to balance the need for accurate data classification while minimising power consumption, as the inclusion of gyroscope data is typically associated with higher power consumption. Table I and Table II show detailed results for each feature dataset. The models achieved an average accuracy between 94.2% and 95.7%, with loss functions between 0.082 and 0.101. The architectures using additional gyroscope data show a slightly lower loss function while maintaining the same accuracy as without a gyroscope. However, in addition to their effectiveness in classifying falls and ADL, they showed a higher rate of false negatives (FN) (13.15% for data including triaxial accelerometer and gyroscope values and 16.57% for data including triaxial accelerometer and gyroscope values along with their respective magnitudes) for near falls, predicting them mainly as ADL. These models were, therefore, excluded from further consideration. Moreover, the impact of incorporating gyroscope data on power consumption was also considered in this decision.

The average accuracy and loss function for acceleration and acceleration with magnitude features are almost identical. However, the dataset, which included acceleration with magnitude, was selected due to a slightly lower loss function and its generalisation over different data splits compared to the dataset, which only contained triaxial acceleration. The model demonstrates robust performance in classifying falls, near falls and ADL, with true positive (TP) rates exceeding 90% for each class. In contrast, the feature set containing only acceleration values shows a significant drop in precision due to incorrect classification of near falls, with only 88.64% TPs, where 5.68% are misclassified as ADL and 5.68% as falls. Continuing with magnitude data could be advantageous for broader applications and additional activities. This decision is based on the understanding that individuals fall in different directions, resulting in fluctuations in the axial sensor values. In contrast, the magnitude remains consistent regardless of the direction of the fall, providing a more stable reference point for analysis.

TABLE I
ACCURACY (A) AND LOSS (L) OF THE CNN-LSTM MODEL USING TIMESERIES DATA, INCLUDING TRIAXIAL ACCELEROMETER VALUES (ACC) AND TRIAXIAL ACCELEROMETER ALONGSIDE THEIR MAGNITUDE (ACC + M).

	ACC		ACC + M	
	A/%	**L**	**A/%**	**L**
16-1 Split	95	0.094	92.6	0.088
80-20 Split	95.1	0.107	96	0.101
Average	95.1	0.101	94.3	0.095

TABLE II
ACCURACY (A) AND LOSS (L) OF THE CNN-LSTM MODEL USING TIMESERIES DATA, INCLUDING TRIAXIAL ACCELEROMETER AND GYROSCOPE VALUES (ACC + GYR) AND TRIAXIAL ACCELEROMETER AND GYROSCOPE VALUES ALONG WITH THEIR RESPECTIVE MAGNITUDES (ACC + GYR + M).

	ACC + GYR		ACC + GYR + M	
	A/%	**L**	**A/%**	**L**
16-1 Split	95.2	0.086	92.5	0.078
80-20 Split	96.1	0.08	95.9	0.085
Average	95.7	0.083	94.2	0.082

B. Performance of hypertuned Machine Learning Algorithms

The results of the performance evaluations for the sensor-based fall detection models, using timeseries data containing triaxial accelerometer values and their magnitudes, are summarised in Table III. The table compares the performance metrics of three machine learning algorithms. The CNN model achieved an average accuracy over both splits of 95.2% and a loss of 0.14. Similarly, the average performance of the LSTM model over both splits is an accuracy of 96.6% and a loss of 0.152. Notably, the hybrid model outperformed the individual CNN and LSTM models, see Figure 3. On the 16-1 split, it achieved an accuracy of 96.4% and a loss of 0.154. On the 80-20 split, it achieved the highest accuracy of 98.8% with a loss of 0.064. The average performance of the CNN-LSTM hybrid model across both splits is an accuracy of 97.6% and a loss of 0.109. The hybrid model uses the strengths of both CNNs and LSTMs. While CNNs excel in capturing spatial patterns, LSTMs are trained to perform well in capturing temporal dependencies.

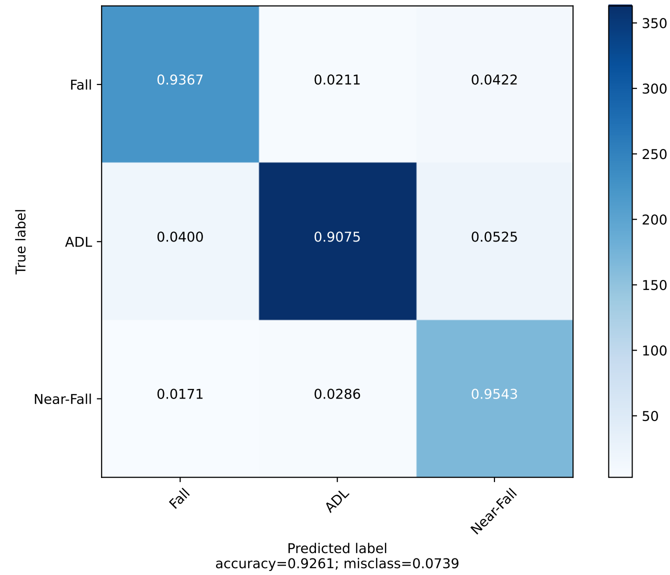

Fig. 2. Confusion matrix of the CNN-LSTM architecture using the 16-1 split of timeseries data, including triaxial accelerometer values with their magnitude.

Based on the accuracy and loss metrics, the CNN-LSTM and LSTM architectures are selected for further development and knowledge distillation to deploy the machine learning algorithm on a microcontroller. The hypertuned CNN model was not further considered because it showed a higher rate of FN (15.43%) for near falls, predicting them mainly as ADL.

TABLE III
ACCURACY (A) AND LOSS (L) OF HYPERTUNED MACHINE LEARNING ALGORITHMS TRAINED USING TIMESERIES DATA, INCORPORATING TRIAXIAL ACCELEROMETER VALUES AND THEIR RESPECTIVE MAGNITUDES.

	CNN		LSTM		CNN-LSTM	
	A/%	**L**	**A/%**	**L**	**A/%**	**L**
16-1 Split	93.7	0.192	95.8	0.191	96.4	0.154
80-20 Split	96.6	0.088	97.3	0.112	98.8	0.064
Average	95.2	0.14	96.6	0.152	97.6	0.109

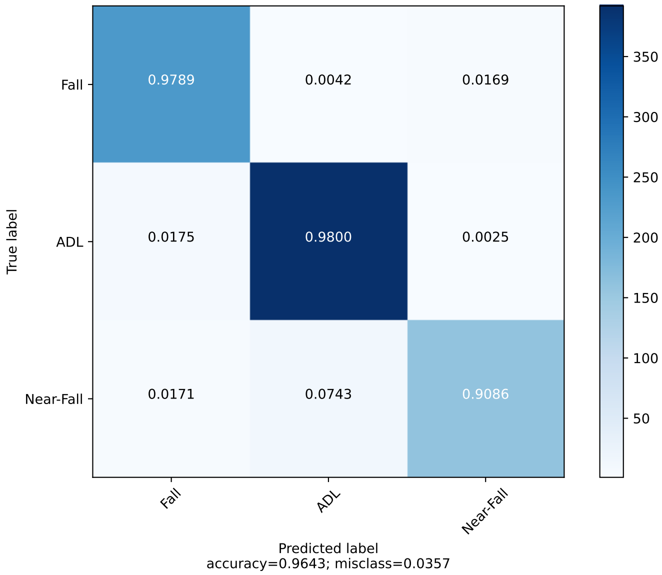

Fig. 3. Confusion matrix of the hypertuned CNN-LSTM architecture, applying the 16-1 split of timeseries data, including triaxial accelerometer values along with their magnitude.

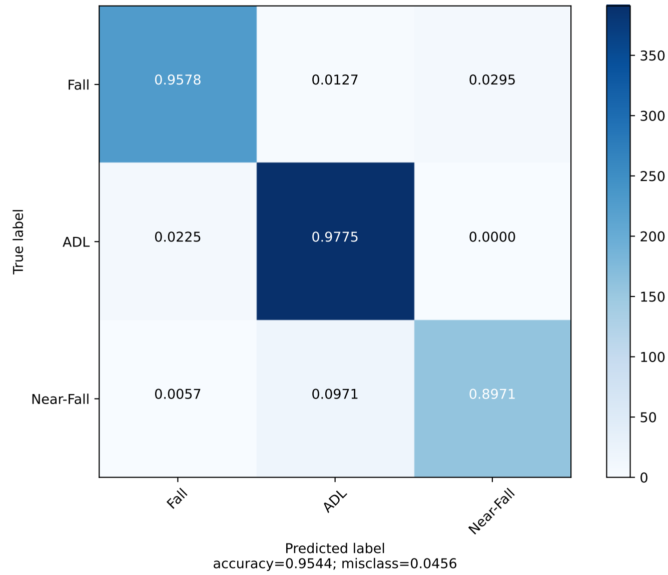

Fig. 4. Confusion matrix of the downscaled low-power CNN-LSTM architecture, applying the 16-1 split of timeseries data, including triaxial accelerometer values along with their magnitude.

C. *Performance of low-power Machine Learning Algorithms*

The performance evaluation results for the downscaled low-power fall detection algorithms are summarised in Table IV. This table provides a comparative analysis of the accuracy, weighted precision, weighted recall and weighted F1-Score metrics for two machine learning algorithms.

The LSTM architecture shows average performance values between 88.6% and 95.6%, while the CNN-LSTM consistently performs with high values between 94.7% and 96.1%. Due to the complexity of LSTM layers compared to convolutional or dense layers, the LSTM teacher model had a memory size of 1.61 MB. Consequently, reducing the model size by 88.5% to 185 kB to approach the desired size goal resulted in decreased performance. In particular, the LSTM model struggled to accurately classify near falls, with 25.17% FN at the 16-1 split, predicting them mainly as ADL. In addition, the loss function yielded a value of 0.268. In comparison, the downscaled CNN-LSTM model maintained high performance metrics despite being reduced by 75.1% to 94 kB and 24301 total parameters. It showed only a 1% reduction in accuracy compared to the teacher model for the 16-1 split, along with a slightly higher loss function of 0.177.

TABLE IV
ACCURACY (A), WEIGHTED PRECISION (P), WEIGHTED RECALL (R) AND WEIGHTED F1-SCORE (F1) OF THE DOWNSCALED LOW-POWER LSTM AND CNN-LSTM ARCHITECTURES, INCORPORATING TRIAXIAL ACCELEROMETER VALUES AND THEIR RESPECTIVE MAGNITUDES.

	LSTM				**CNN-LSTM**			
	A/%	**P/%**	**R/%**	**F1/%**	**A/%**	**P/%**	**R/%**	**F1/%**
16-1 Split	90.6	88.6	90.7	89.2	95.4	93.5	95.3	94.2
80-20 Split	95.6	95.6	95.6	95.6	96.8	95.8	96.8	96.2
Average	93.1	92.1	93.2	92.4	96.1	94.7	96.1	95.2

IV. DISCUSSION

A. *Acceleration Data with Magnitude for Improved Classification*

Several factors drove the decision to use triaxial acceleration data, along with their respective magnitudes, as features for classification. This choice was influenced by classification effectiveness, stability and power efficiency considerations, making it suitable for use in MCUs. In contrast, including gyroscope data alongside accelerometer data increases power consumption due to additional sensor readings and processing but does not perform better.

Using acceleration data with magnitude provides a broad representation of motion dynamics independent of the direction of motion. This invariant representation simplifies classification by focusing on overall motion rather than direction-specific patterns. Magnitude data provides a stable reference point for analysis and is immune to orientation changes or sensor noise. This stability enhances the model's ability to differentiate between falls, near falls and ADL, even under challenging conditions or noisy sensor environments. The stability of the feature representation also contributes to the model's adaptability to different scenarios and user demographics. It ensures that the model remains effective despite user age, height and mobility differences that can affect movement characteristics.

B. *CNN-LSTM Architecture for Timeseries Data Classification*

The choice of a CNN-LSTM architecture for classifying timeseries data into falls, near falls and ADL was based on several crucial advantages observed during the investigation. This architecture combines the strengths of CNNs in capturing spatial patterns and LSTM networks in modelling temporal dependencies. Falls and near falls show clearer spatial patterns and temporal dynamics than ADL. CNNs effectively capture spatial features from raw sensor data, which is essential for identifying specific movement patterns. On the other hand, LSTM networks are effective at modelling the temporal dependencies present in timeseries data, enabling the detection of motion sequences indicative of falls or near falls.

The CNN-LSTM architecture demonstrated outstanding performance in terms of accuracy and loss compared to either CNN or LSTM models alone. This improved performance is related to the collaboration of the CNN and LSTM layers, which use spatial and temporal information for accurate classification. In addition, the flexibility of the CNN-LSTM architecture allows it to adapt to different timeseries data, making it suitable for a wide range of applications beyond fall detection.

V. Conclusion

The CNN-LSTM architecture provides a robust framework for classifying timeseries data into fall, near fall and ADL categories. This architecture achieves high performance in classification tasks by combining spatial and temporal information compared to the standalone models. In addition, using acceleration data with magnitude improves classification accuracy while maintaining performance efficiency. These findings contribute to the development of an effective and efficient fall detection system implemented in hearing implant systems, thus improving the safety and well-being of older adults and enabling them to maintain their independence and autonomy in daily life.

The results provide promising insights into the effectiveness and efficiency of the system. However, further research is essential to improve the system to the point where it can be implemented in a market-ready device. In particular, additional near fall timeseries data is needed to improve the classification of near falls or to classify into falls and no falls, as mainly all FN classified as ADL are near falls. Combining near falls and ADL into a single class would reduce the loss function and increase the accuracy of the fall detection algorithm. This solution would allow the algorithm to be even smaller in memory size without any performance loss.

Acknowledgment

I am deeply thankful to MED-EL Medical Electronics for their support and resources and to my supervisors, FH-Prof. Dr. Dipl.-Ing. Daniel Sieber and Philipp Schmidt, MSc., for their guidance and feedback. Special appreciation goes to the team at MED-EL Medical Electronics for their support and to my friends and family for their encouragement.

References

[1] W. H. Organization, "World report on ageing and health," *Ageing And Health*, 2015.

[2] U. Nations, "World population prospects the 2015 revision," *World Population Prospects*, 2015. [Online]. Available: http://esa.un.org/unpd/wpp/Publications/Files/WPP2015Volume-IComprehensive-Tables.pdf.

[3] P. Becker, J. Schütz, and A. Zimmermann, "The network of europes leading demographic research centres ageing workforce, social cohesion and sustainable development political challenges within the baltic sea region," *Population Europe*, 2018.

[4] J. A. Stevens, P. S. Corso, E. A. Finkelstein, and T. R. Miller, "The costs of fatal and non-fatal falls among older adults," *Journal of the International Society for Child and Adolescent Injury Prevention*, vol. 12, pp. 290–295, 10 2006. [Online]. Available: http://dx.doi.org/10.1136/IP.2005.011015

[5] W. H. Organization, "Step safely - strategies for preventing and managing falls across the life-course," *Step Safely*, pp. 4–38, 2021.

[6] D. A. Sterling, J. A. O'Connor, and J. Bonadies, "Geriatric falls: injury severity is high and disproportionate to mechanism," *The Journal of trauma*, vol. 50, pp. 116–119, 2001. [Online]. Available: http://dx.doi.org/10.1097/00005373-200101000-00021

[7] N. S. Reed, E. E. Garcia-Morales, C. Myers, A. R. Huang, J. R. Ehrlich, O. J. Killeen, J. E. Hoover-Fong, F. R. Lin, M. L. Arnold, E. S. Oh, J. A. Schrack, and J. A. Deal, "Prevalence of hearing loss and hearing aid use among us medicare beneficiaries aged 71 years and older," *JAMA Network Open*, vol. 6, 7 2023. [Online]. Available: http://dx.doi.org/10.1001/JAMANETWORKOPEN.2023.26320

[8] N. T. Newaz and E. Hanada, "The methods of fall detection: A literature review," *Sensors 2023, Vol. 23, Page 5212*, vol. 23, p. 5212, 5 2023. [Online]. Available: http://dx.doi.org/10.3390/S23115212

[9] F. Harrou, N. Zerrouki, Y. Sun, and A. Houacine, "Statistical control chart and neural network classification for improving human fall detection," in *2016 8th International Conference on Modelling, Identification and Control (ICMIC)*. IEEE, 2016, pp. 1060–1064. [Online]. Available: http://dx.doi.org/10.1109/icmic.2016.7804269

[10] A. Núñez-Marcos, G. Azkune, I. Arganda-Carreras *et al.*, "Vision-based fall detection with convolutional neural networks," *Wireless communications and mobile computing*, vol. 2017, 2017. [Online]. Available: https://doi.org/10.1155/2017/9474806

[11] E. Alam, A. Sufian, P. Dutta, and M. Leo, "Vision-based human fall detection systems using deep learning: A review," *Computers in Biology and Medicine*, vol. 146, 7 2022. [Online]. Available: https://doi.org/10.1016/j.compbiomed.2022.105626

[12] S. Nooruddin, M. M. Islam, F. A. Sharna, H. Alhetari, and M. N. Kabir, "Sensor-based fall detection systems: a review," *Journal of Ambient Intelligence and Humanized Computing*, vol. 13, pp. 2735–2751, 5 2022. [Online]. Available: https://doi.org/10.1007/S12652-021-03248-Z

[13] X. Wang and K. Jia, "Human fall detection algorithm based on yolov3," in *2020 IEEE 5th International Conference on Image, Vision and Computing (ICIVC)*. IEEE, 2020, pp. 50–54. [Online]. Available: https://doi.org/10.1109/ICIVC50857.2020.9177447

[14] T. Lee and A. Mihailidis, "An intelligent emergency response system: preliminary development and testing of automated fall detection," *Journal of telemedicine and telecare*, vol. 11, no. 4, pp. 194–198, 2005. [Online]. Available: https://doi.org/10.1258/1357633054068946

[15] C. Wang, L. Tang, M. Zhou, Y. Ding, X. Zhuang, and J. Wu, "Indoor human fall detection algorithm based on wireless sensing," *Tsinghua Science and Technology*, vol. 27, no. 6, pp. 1002–1015, 2022. [Online]. Available: https://doi.org/10.26599/TST.2022.9010011

[16] 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, no. 3, pp. 489–501, 2014. [Online]. Available: https://doi.org/10.1016/j.cmpb.2014.09.005

[17] M. Vallejo, C. V. Isaza, and J. D. Lopez, "Artificial neural networks as an alternative to traditional fall detection methods," in *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2013, pp. 1648–1651. [Online]. Available: https://doi.org/10.1109/EMBC.2013.6609833
[18] A. Sengto and T. Leauhatong, "Human falling detection algorithm using back propagation neural network," in *The 5th 2012 Biomedical Engineering International Conference*. IEEE, 2012, pp. 1–5. [Online]. Available: https://doi.org/10.1109/BMEiCon.2012.6465460
[19] Y. Li, Z. Zuo, and J. Pan, "Sensor-based fall detection using a combination model of a temporal convolutional network and a gated recurrent unit," *Future Generation Computer Systems*, vol. 139, pp. 53–63, 2023. [Online]. Available: https://doi.org/10.1016/j.future.2022.09.011
[20] J. R. Burwinkel, B. Xu, and J. Crukley, "Preliminary examination of the accuracy of a fall detection device embedded into hearing instruments," *Journal of the American Academy of Audiology*, vol. 31, no. 06, pp. 393–403, 2020. [Online]. Available: https://doi.org/10.3766/jaaa.19056
[21] J. R. Burwinkel and B. Xu, "Livio edge ai detects falls accurately and alerts others," *Future Generation Computer Systems*, 2020.
[22] A. Zdemir and B. Barshan, "Simulated Falls and Daily Living Activities Data Set," UCI Machine Learning Repository, 2018. [Online]. Available: https://doi.org/10.24432/C52028
[23] T. O'Malley, E. Bursztein, J. Long, F. Chollet, H. Jin, L. Invernizzi *et al.*, "Kerastuner," 2019.
[24] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," 2015. [Online]. Available: https://doi.org/10.48550/arXiv.1503.02531

Fabian Schießendobler is with the Department of Medical and Health Technologies, MCI, Innsbruck, Austria. Among others he is responsible for the activities in medical engineering in which he is regularly publishing.