

Time-Distributed Layer Convolutions with Long Short-Term Memory for Human Activity Recognition and Result Comparison with Various Machine Learning Models

Jaykumar S. Dhage

Dr. B.A.M. University, Aurangabad, India
jaykumardhage2021@gmail.com

Avinash K. Gulve

Government Engineering College, Aurangabad, India
akgulve@yahoo.com (corresponding author)

Received: 7 February 2025 | Revised: 10 March 2025, 25 March 2025, and 29 March 2025 | Accepted: 30 March 2025

Licensed under a CC-BY 4.0 license | Copyright (c) by the authors | DOI: <https://doi.org/10.48084/etasr.10499>

ABSTRACT

Human Activity Recognition (HAR) is an essential area of research with many applications in healthcare, security, and entertainment. One of the main challenges in HAR is the variability in human behavior and reactions to similar inputs, which complicates accurate prediction. This study investigates the utilization of deep learning techniques in enhancing the HAR accuracy. The proposed method uses a Time-Distributed Layer (TDL) framework with LSTM to achieve effective feature extraction and temporal pattern recognition from sensor data. The proposed approach was compared with traditional machine learning models, such as Logistic Regression (LR), Support Vector Machines (SVM), Decision Trees (DT), and Random Forests (RF), to evaluate its effectiveness. The experimental results demonstrate that deep learning models significantly outperform traditional approaches, achieving 97.57% accuracy with TDL-LSTM and 97.81% accuracy with LSTM-TDL, while conventional methods exhibit lower performance. The comparison highlights the advantages of deep learning methods in capturing both spatial and temporal dependencies, resulting in more robust HAR systems. Overall, this study demonstrates the superiority of LSTM-based architectures over traditional models, paving the way for future advances in real-world HAR applications, including wearable devices and intelligent monitoring systems.

Keywords-Human Activity Recognition (HAR); Time Distributed layer (TDL); Long Short-Term Memory (LSTM)

I. INTRODUCTION

Today, one of the most crucial areas for research is the detection and identification of human activities [1]. Detecting human activities is now easier since almost everyone has a smartphone. A variety of sensors available in smartphones, such as accelerometers, gyroscopes, microphones, and cameras, help to extract data to accurately recognize activities [2, 3]. These devices are low-cost and require very little energy to work. Human Activity Recognition (HAR) has a wide range of applications, such as healthcare, social networks, safety, detection of suspicious activities, transportation, and surveillance systems [4]. In addition, it can distinguish the activities of younger and older people [5]. Artificial Intelligence (AI) and the Internet of Things (IoT) help in HAR. Due to these new components, HAR is becoming more complex and interesting and opens a new topic for research [6].

HAR has witnessed a transformative surge, propelled by the rapid advancements in deep learning techniques such as Long Short-Term Memory (LSTM) networks and Convolutional Neural Networks (CNNs) [7]. These sophisticated algorithms have revolutionized the ability to interpret and predict human behaviors from sensor data with unprecedented accuracy and efficiency. At the heart of this revolution lies LSTM, a type of Recurrent Neural Network (RNN) designed to capture long-term dependencies in sequential data. Unlike traditional RNNs, LSTM networks are equipped with a memory cell that can maintain information over extended periods, making them particularly effective for tasks that require understanding temporal dynamics in human activities [8]. Complementing LSTM, CNNs have proven invaluable in extracting spatial features from sensor data. Initially developed for image recognition tasks, CNNs have been effectively adapted to analyze spatial patterns in sensor readings, enhancing the robustness and versatility of HAR systems [9]. Among the

intense adoption of deep learning approaches, a diverse array of machine learning algorithms further enriches the landscape of HAR. From Decision Trees (DT) to Support Vector Machines (SVM), each algorithm brings distinct strengths in modeling complex relationships within sensor data, catering to various application domains and dataset characteristics [10-14].

HAR is evolving rapidly with the integration of deep learning techniques such as LSTM and CNNs alongside traditional machine learning models. However, accurate recognition remains a challenge due to variations in movement patterns, sensor noise, and the need for efficient feature extraction. Traditional approaches such as SVM, DT, and RF rely on handcrafted features, which limits their adaptability. This study introduces a novel TDL-LSTM framework, which enhances spatiotemporal feature learning for improved activity classification. Unlike previous models that process temporal and spatial features separately, this approach effectively captures sequential dependencies, leading to higher accuracy compared to existing models such as 1D CNN and SVM. A detailed comparative analysis with both traditional and deep learning models confirms its superior performance. In addition, the proposed model is optimized for real-world applications, ensuring efficiency in healthcare monitoring, security, and wearable technology. This research advances HAR by bridging deep learning innovations with statistical methods, offering a scalable and high-accuracy solution for diverse environments.

II. LITERATURE SURVEY

HAR has been extensively studied using both traditional machine learning and deep learning approaches. Although previous studies have contributed significantly to the advancement of HAR, they exhibit certain limitations that this study aims to address. In [9], a 1D CNN-based approach was proposed for HAR, achieving 96.13% accuracy on the UCI dataset. However, 1D CNNs struggle with long-term dependencies in sequential data, making them less effective in capturing complex activity transitions. In [14-16] 1D CNN models were improved with data augmentation, improving accuracy while reducing false positives and computational complexity. However, these methods still rely heavily on preprocessed features and lack end-to-end learning capabilities. In [1], an SVM with an RBF kernel achieved 98.96% accuracy for three activity labels. However, although this approach performed well, its limited scope of activity classification and its dependence on feature engineering restrict its applicability to real-world scenarios [13, 14]. In [17], CNN was combined with Graph Convolutional Networks (GCN) to improve performance, but graph-based approaches require extensive computational resources, making them less efficient for real-time applications. In [18], CNNs were used on accelerometer data, obtaining 92.71% accuracy for three activity classes. However, CNNs alone fail to capture temporal dependencies in activity sequences, leading to reduced accuracy in motion transition detection.

This study addresses these limitations by introducing a novel TDL-LSTM framework that integrates time-distributed convolution with LSTM-based sequential learning. Unlike previous CNN-based or ML-based models, this approach effectively captures both spatial and temporal dependencies

while minimizing computational complexity and feature engineering requirements. The results demonstrate superior accuracy (97.81%), highlighting the potential of this approach in real-world HAR applications.

III. GENERAL ARCHITECTURE

Figure 1 shows the HAR process flow. This study used the UCI dataset [5, 18-22]. The data were collected from a group of 30 volunteers. The integrated gyroscope and accelerometer assisted in recording the three-axial angular velocity and three-axial linear acceleration. This dataset uses six activity labels. All the results are labeled videos. The dataset was divided into 70% for training and the remaining 30% for testing. Filters were applied to preprocess sensor signals. Next, sliding windows that last 2.56 seconds and overlap 50% were used to sample them. A feature vector was obtained from each window by calculating variables from the two domains, including (i) Triaxial acceleration and body acceleration, (ii) Triaxial angular velocity, (iii) 561 attributes (iv) Six activity labels. (0: Laying, 1: Sitting, 2: Standing, 3: Walking, 4: Walking Downstairs, 5: Walking Upstairs), (v) An identifier for the subject.

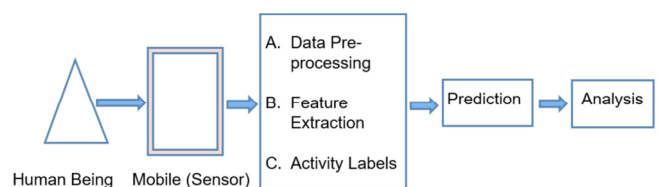


Fig. 1. The flow of process for HAR.

The Butterworth filter is used to remove noise. Linear acceleration and angular velocity are collected to obtain Jerk signals. The Euclidean norm is used to determine the signals' magnitude. This allows the variables for each pattern to be estimated. The symbol indicates three-directional axial signals. Various optimizer methods, such as FTRL, Adam, RMSprop, and Gradient Descent exist. In addition to the Gradient Descent algorithm, many other algorithms have been proposed. However, it is the king of all optimizers and is challenging, fast, and adjustable. This optimizer calculates any slight change in each weight component that affects the loss function. Based on its gradient, the tuning of each weight is achieved. These steps continue until the loss function reaches its minimum. The RMSprop optimizer is one version of AdaGrad that accumulates gradients only in a specific fixed window rather than doing it for all. Like the RMSprop optimizer, the Adadelta optimizer is another enhanced optimization technique where the current and newly updated weight differences are stored in delta. Adam stands for adaptive moment estimation, which uses the concept of momentum. Here, the addition of fractions of a current and a back node is considered. The Stochastic Gradient Descent (SGD) optimization algorithm performs complex computations for larger datasets by pre-calculating gradients before any updates. It performs frequent updates with high variance, resulting in high fluctuations in objective functions.

IV. DEEP LEARNING APPROACH

For time range data, such as stock prices and video frames, LSTM is a suitable approach since it selectively remembers patterns for long periods. LSTM has cell states with different dependencies that do not manipulate all the information but modify it slightly. Sequence categorization relies heavily on LSTMs due to their ability to learn directly from the raw time series data. They do not require any prior subject-matter expertise to manually develop input features. The TDL approach can help to adopt each input before or after this LSTM layer. Keras provides an excellent solution, offering frame-by-frame activities and consecutive and sequential actions. Therefore, with TDL and LSTM, this model can provide very accurate results, provided that fine synchronization is achieved.

TABLE I. CONVOLUTION ARCHITECTURE: TDL AND THEN LSTM

Layer	Output pattern	Parameters
TD	TD (None, None, 30, 64)	1792
TD	TD (None, None, 28, 64)	12352
TD	TD ((None, None, 28, 64)	0
TD	TD (None, None, 14, 6 4)	0
TD	TD (None, None, 896)	0
LSTM	(None, 32)	118912
Dropout	(None, 32)	0
Dense	(None, 32)	1056
Dense	(None, 6)	198

In this case, the total parameters for training are 134,310. The Time-Distributed (TD) layer works with several inputs and produces one output per input to get the result in time. Table I shows the convolution architecture of TDL and LSTM. In HAR, an object must be checked in motion. Therefore, searching for the object is more necessary before detecting the movement. That is why this model needs to make convolutions before LSTM. The distributed layer is robust in that, irrespective of its position with LSTM, the effect on the data will be almost the same. Only one model can do the work. Keras is used to make movement prediction and recognition. A series of operations, as shown in Figure 2, are performed using the given input [17, 23-24].

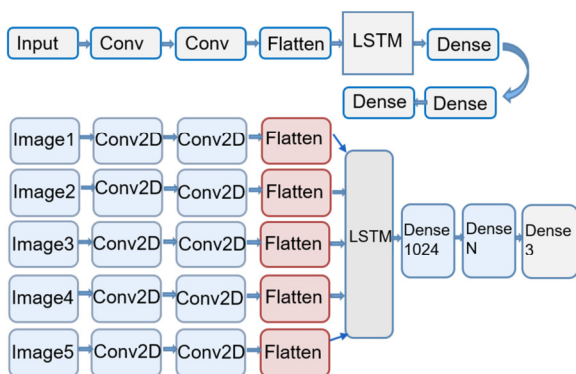


Fig. 2. Steps for the convolution process (TDL and then LSTM).

Table II outlines the structure and specifications of an LSTM-TDL, detailing various layers and their respective

output patterns and parameter counts. The model primarily utilizes LSTM layers for sequence processing, starting with an output of shape (None, 128, 32) with 5,376 parameters. This is followed by several TD layers, with configurations yielding outputs of (None, 128, 128) and (None, 128, 64), contributing 4,224 and 8,256 parameters, respectively. The model then transitions to a Flatten layer that reshapes the output into a single vector of size (None, 8192), leading into a Dense layer with 524,352 parameters designed to produce 64 output units. A Dropout layer follows to prevent overfitting, before concluding with a final Dense layer that outputs 6 classes, accounting for 390 parameters. In general, this architecture illustrates a complex configuration designed for effective learning from sequential data.

TABLE II. CONVOLUTION ARCHITECTURE: LSTM AND THEN TDL

Layer	Output pattern	Parameters
(LSTM)	(None, 128, 32)	5376
TD	TD (None, 128, 128)	4224
TD	TD ((None, 128, 64)	8256
Flatten	(None, 8192)	0
Dense	(None, 64)	524352
Dropout	(None, 64)	0
Dense	(None, 6)	390

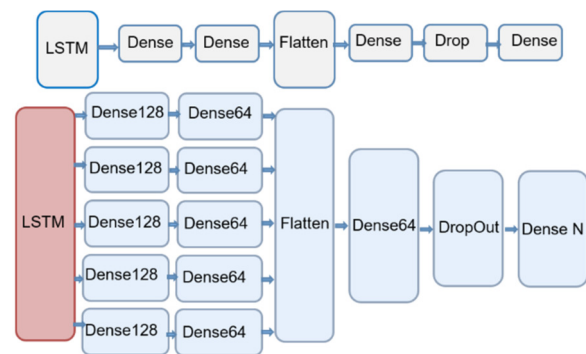


Fig. 3. Steps for the convolution process (LSTM and then TDL).

In this case, the total parameters for training are 542,598.

Deep learning models with large parameter sets are prone to overfitting, especially with limited or imbalanced datasets. To mitigate this, dropout layers and batch normalization were incorporated to enhance generalization and prevent reliance on specific neurons. Additionally, early stopping is used during training to halt the process when validation loss increases, reducing overfitting risks. Data augmentation techniques further help in increasing dataset diversity, while a carefully balanced 70-30% dataset split ensures a representative distribution of activity classes. These strategies collectively improve the robustness and adaptability of the model, making it more effective for real-world HAR applications.

A series of operations are performed with the given input, as shown in Figure 3. The figure illustrates a neural network architecture featuring an LSTM layer followed by a series of fully connected (dense) layers. The LSTM layer processes sequential data, outputting features fed into four parallel streams, each comprising two dense layers with 128 and 64

units, respectively. These streams converge into a flattening layer, which converts the multi-dimensional input into a single dimension. The flattened output is then processed by an additional dense layer with 64 units, followed by a dropout layer that helps prevent overfitting by randomly setting a fraction of input units to zero during training. Finally, the output passes through a dense layer with 6 units, representing the final predictions of the model. This architecture is designed to capture temporal dependencies and complex patterns in sequential data effectively.

V. RESULTS AND DISCUSSION

The standard dataset has a total of 10,299 records. For training purposes, 70% of the data, i.e., 7,352 records, were

used, whereas 30% of the data, i.e., 2,947 records, were used for testing purposes. As shown in Figure 4(a, b, c, d, e), the LR, SVM, DT, RF, and NN models achieved the least accurate results for the Walking Downstairs label and the most accurate results for the Laying label. The LR, SVM, DT, RF, and NN models achieved accuracies of 95.82%, 96.30%, 85.20%, 92.05%, and 95.07%, respectively. Additional performance metrics, such as precision, F1 score, and recall, were also considered. As shown in Figure 4(f, g), the most accurate results were obtained when TDL was applied. Using TDL before LSTM achieves an accuracy of 97.57%, and LSTM before TDL achieves an accuracy of 97.81%. Both these models achieved their best prediction on Walking Upstairs and their lowest prediction on Walking.

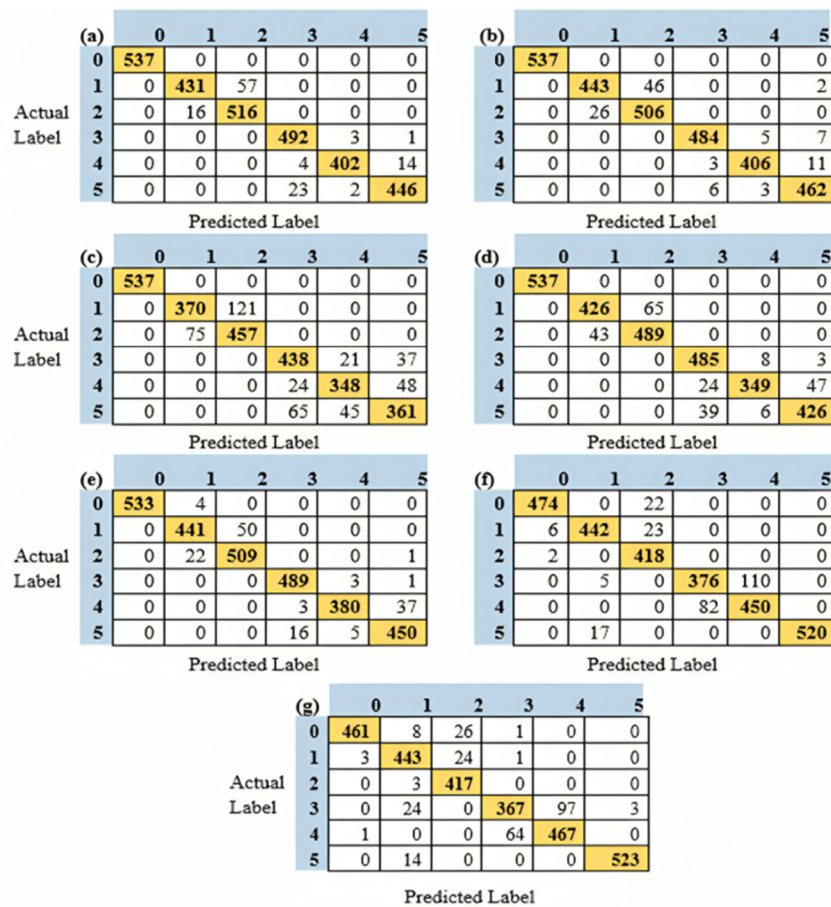


Fig. 4. Confusion matrix of (a) LR, (b) SVM, (c) DT, (d) RF, (e) NN, (f) TDL → LSTM, (g) LSTM → TDL.

Table III presents predictions from various algorithms for different activity labels, illustrating the most and least frequently predicted samples per label. All algorithms showed uniform predictions on the Laying label, with counts between 533 and 537. However, the Walking Downstairs label had more fluctuations, with predictions ranging from 348 to 406. These variations indicate strengths and weaknesses in the performance of each algorithm. In particular, the DT and RF algorithms yielded fewer predictions for the Walking Downstairs label, suggesting areas that need improvement, while the NN displayed a distinct prediction pattern. The TDL

to LSTM transition showed greater consistency in predicting Walking Upstairs than vice versa. Analyzing these discrepancies is essential to improve algorithm efficacy and improve prediction accuracy in future applications. Table IV presents the performance metrics of various algorithms, including accuracy, precision, recall, and F1 score. In particular, the LSTM to TDL method achieved the highest accuracy at 97.81%, closely followed by the TDL to LSTM at 97.57%, demonstrating their effectiveness compared to traditional algorithms such as LR and SVM.

TABLE III. CONFUSION MATRIX STATISTICS OF VARIOUS ALGORITHMS

	Algorithm	Activity label	Samples predicted (Most)	Activity label	Samples predicted (Least)
1	LR	Laying	537	Walking Downstairs	402
2	SVM	Laying	537	Walking Downstairs	406
3	DT	Laying	537	Walking Downstairs	348
4	RF	Laying	537	Walking Downstairs	349
5	NN	Laying	533	Walking Downstairs	380
6	TDL→LSTM	Walking Upstairs	520	Walking	376
7	LSTM→TDL	Walking Upstairs	523	Walking	367

TABLE IV. PERFORMANCE OF VARIOUS ALGORITHMS

	Algorithm	Accuracy	Precision	Recall	F1 score
1	LR	95.82	0.956	0.94	0.94
2	SVM	96.30	0.96	0.95	0.95
3	DT	85.20	0.86	0.85	0.85
4	RF	92.05	0.92	0.91	0.91
5	NN	95.07	0.94	0.93	0.93
6	TDL→LSTM	97.57	0.95	0.95	0.96
7	LSTM→TDL	97.81	0.96	0.97	00.97

Figure 5 shows a comparative overview of various machine learning and deep learning algorithms, highlighting the performance of models such as LR, SVM, DT, RF, NN, and the combination of TDL with LSTM. Each algorithm has unique strengths and applications, ranging from traditional approaches such as LR and DT to more complex architectures such as NN and LSTMs. This diversity in methods allows practitioners to choose the most suitable model based on the specific characteristics of their datasets and the nature of the problems being addressed.

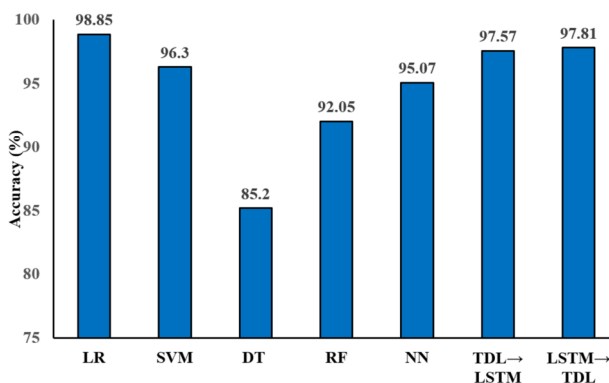


Fig. 5. Graph of accuracy results.

Table V provides a comparative analysis of this study with earlier works on HAR. The proposed deep learning models (TDL-LSTM and LSTM-TDL) show higher accuracy (97.57% and 97.81%) compared to traditional machine learning and previous deep learning-based approaches.

TABLE V. COMPARATIVE ANALYSIS WITH PREVIOUS STUDIES

Study	Method Used	Dataset used	Accuracy
[1]	SVM with RBF Kernel	HAR	98.96%
[9]	1D CNN	UCI HAR	96.13%
[18]	CNN-based accelerometer data processing	Accelerometer data	92.71%
This study	TDL with LSTM	UCI HAR	97.57%
This study	LSTM with TDL	UCI HAR	97.81%

VI. CONCLUSION

This study presents a novel approach to HAR by integrating TDL and LSTM networks, achieving superior accuracy in activity classification. Unlike traditional machine learning methods, such as LR, SVM, DT, and RF, which depend on manual feature extraction, this deep learning model autonomously learns complex spatial and temporal patterns from sensor data. The proposed TDL-LSTM and LSTM-TDL architectures achieved outstanding accuracies of 97.57% and 97.81%, respectively, outperforming conventional models. A key contribution of this work is the efficient fusion of convolutional processing with LSTM-based sequential learning, allowing for precise recognition of subtle activity differences, such as walking upstairs versus downstairs. Compared to previous HAR models [9, 10], these models provide higher generalized accuracy across multiple activity classes. In addition, this study addresses computational efficiency by reducing the need for handcrafted feature engineering, making it adaptable for real-world applications in healthcare, security, and wearable devices. Future research can explore hybrid deep learning frameworks to further optimize real-time HAR systems.

DECLARATIONS

Data availability of data and materials: The UCI dataset was obtained from [22].

Funding: This work has not received any funding from any source.

REFERENCES

- [1] A. E. Minarno, W. A. Kusuma, and H. Wibowo, "Performance Comparison Activity Recognition using Logistic Regression and Support Vector Machine," in *2020 3rd International Conference on Intelligent Autonomous Systems (ICoIAS)*, Singapore, Feb. 2020, pp. 19–24, <https://doi.org/10.1109/ICoIAS49312.2020.9081858>.
- [2] A. Hayat, F. Morgado-Dias, B. P. Bhuyan, and R. Tomar, "Human Activity Recognition for Elderly People Using Machine and Deep Learning Approaches," *Information*, vol. 13, no. 6, Jun. 2022, Art. no. 275, <https://doi.org/10.3390/info13060275>.
- [3] A. Bulling, U. Blanke, and B. Schiele, "A tutorial on human activity recognition using body-worn inertial sensors," *ACM Computing Surveys*, vol. 46, no. 3, Jan. 2014, Art. no. 31, <https://doi.org/10.1145/2499621>.
- [4] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "Human Activity Recognition on Smartphones Using a Multiclass Hardware-Friendly Support Vector Machine," in *Ambient Assisted Living and Home Care*, 2012, pp. 216–223, https://doi.org/10.1007/978-3-642-35395-6_30.
- [5] D. Anguita, A. Ghio, L. Oneto, X. Parra Perez, and J. L. Reyes Ortiz, "A public domain dataset for human activity recognition using smartphones," presented at the Proceedings of the 21th International European Symposium on Artificial Neural Networks, Computational

- Intelligence and Machine Learning, 2013, pp. 437–442. [Online]. Available: <https://upcommons.upc.edu/handle/2117/20897>.
- [6] D. Anguita, A. Ghio, L. Oneto, F. X. Llanas Parra, and J. L. Reyes Ortiz, "Energy efficient smartphone-based activity recognition using fixed-point arithmetic," *Journal of universal computer science*, vol. 19, no. 9, pp. 1295–1314, 2013.
- [7] J. L. Reyes Ortiz, A. Ghio, D. Anguita, X. Parra Perez, J. Cabestany Moncusí, and A. Català Mallofré, "Human activity and motion disorder recognition: towards smarter interactive cognitive environments," presented at the Proceedings of the 21th International European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, 2013, pp. 403–412. [Online]. Available: <https://upcommons.upc.edu/handle/2117/20901>.
- [8] K. Chen, D. Zhang, L. Yao, B. Guo, Z. Yu, and Y. Liu, "Deep Learning for Sensor-based Human Activity Recognition: Overview, Challenges, and Opportunities," *ACM Computing Surveys*, vol. 54, no. 4, Feb. 2021, Art. no. 77, <https://doi.org/10.1145/3447744>.
- [9] K. Muralidharan, R. Anirudh, G. Rithvik, P. Saket, A. A. Reghunaath, and M. P. Gopinath, "1D Convolution approach to human activity recognition using sensor data and comparison with machine learning algorithms," *International Journal of Cognitive Computing in Engineering*, vol. 2, pp. 130–143, Jun. 2021, <https://doi.org/10.1016/j.ijcce.2021.09.001>.
- [10] L. B. Marinho, A. H. de Souza Junior, and P. P. Rebouças Filho, "A New Approach to Human Activity Recognition Using Machine Learning Techniques," in *Intelligent Systems Design and Applications*, Cham, 2017, pp. 529–538, https://doi.org/10.1007/978-3-319-53480-0_52.
- [11] L. Xu, W. Yang, Y. Cao, and Q. Li, "Human activity recognition based on random forests," in *2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, Guilin, Jul. 2017, pp. 548–553, <https://doi.org/10.1109/FSKD.2017.8393329>.
- [12] N. Dua, S. N. Singh, V. B. Semwal, and S. K. Challa, "Inception inspired CNN-GRU hybrid network for human activity recognition," *Multimedia Tools and Applications*, vol. 82, no. 4, pp. 5369–5403, Feb. 2023, <https://doi.org/10.1007/s11042-021-11885-x>.
- [13] R. Liu, A. A. Ramli, H. Zhang, E. Henricson, and X. Liu, "An Overview of Human Activity Recognition Using Wearable Sensors: Healthcare and Artificial Intelligence," in *Internet of Things – ICIOT 2021, 2022*, https://doi.org/10.1007/978-3-030-96068-1_1.
- [14] S. Kiranyaz, M. Zabihi, A. B. Rad, T. Ince, R. Hamila, and M. Gabbouj, "Real-time phonocardiogram anomaly detection by adaptive 1D Convolutional Neural Networks," *Neurocomputing*, vol. 411, pp. 291–301, Oct. 2020, <https://doi.org/10.1016/j.neucom.2020.05.063>.
- [15] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, and D. J. Inman, "1D convolutional neural networks and applications: A survey," *Mechanical Systems and Signal Processing*, vol. 151, Apr. 2021, Art. no. 107398, <https://doi.org/10.1016/j.ymsp.2020.107398>.
- [16] S. Kiranyaz, T. Ince, O. Abdeljaber, O. Avci, and M. Gabbouj, "1-D Convolutional Neural Networks for Signal Processing Applications," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Brighton, United Kingdom, May 2019, pp. 8360–8364, <https://doi.org/10.1109/ICASSP.2019.8682194>.
- [17] Y. D. Zhang, S. C. Satapathy, D. S. Guttery, J. M. Górriz, and S. H. Wang, "Improved Breast Cancer Classification Through Combining Graph Convolutional Network and Convolutional Neural Network," *Information Processing & Management*, vol. 58, no. 2, Mar. 2021, Art. no. 102439, <https://doi.org/10.1016/j.ipm.2020.102439>.
- [18] S. M. Lee, S. M. Yoon, and H. Cho, "Human activity recognition from accelerometer data using Convolutional Neural Network," in *2017 IEEE International Conference on Big Data and Smart Computing (BigComp)*, Jeju Island, South Korea, Feb. 2017, pp. 131–134, <https://doi.org/10.1109/BIGCOMP.2017.7881728>.
- [19] S. Sengupta *et al.*, "A review of deep learning with special emphasis on architectures, applications and recent trends," *Knowledge-Based Systems*, vol. 194, Apr. 2020, Art. no. 105596, <https://doi.org/10.1016/j.knsys.2020.105596>.
- [20] S. Wan, L. Qi, X. Xu, C. Tong, and Z. Gu, "Deep Learning Models for Real-time Human Activity Recognition with Smartphones," *Mobile Networks and Applications*, vol. 25, no. 2, pp. 743–755, Apr. 2020, <https://doi.org/10.1007/s11036-019-01445-x>.
- [21] S. Dargan, M. Kumar, M. R. Ayyagari, and G. Kumar, "A Survey of Deep Learning and Its Applications: A New Paradigm to Machine Learning," *Archives of Computational Methods in Engineering*, vol. 27, no. 4, pp. 1071–1092, Sep. 2020, <https://doi.org/10.1007/s11831-019-09344-w>.
- [22] D. A. Jorge Reyes-Ortiz, "Human Activity Recognition Using Smartphones." UCI Machine Learning Repository, 2013, <https://doi.org/10.24432/C54S4K>.
- [23] Y. Liu, S. Liu, Y. Wang, F. Lombardi, and J. Han, "A Stochastic Computational Multi-Layer Perceptron with Backward Propagation," *IEEE Transactions on Computers*, vol. 67, no. 9, pp. 1273–1286, Sep. 2018, <https://doi.org/10.1109/TC.2018.2817237>.
- [24] N. H. Duong, M. T. Nguyen, T. H. Nguyen, and T. P. Tran, "Application of Seasonal Trend Decomposition using Loess and Long Short-Term Memory in Peak Load Forecasting Model in Tien Giang," *Engineering, Technology & Applied Science Research*, vol. 13, no. 5, pp. 11628–11634, Oct. 2023, <https://doi.org/10.48084/etasr.6181>.