A Comparative Analysis of Machine Learning Models for Gym Activity Recognition on an Imbalanced Dataset

Dhruv Kukadiya
School of Computing Science and Engineering
VIT Bhopal University,Kothrikalan,
Sehore, Madhya Pradesh - 466114, India
kukadiya.24mas10011@vitbhopal.ac.in

Trapti Sharma
School of Computing Science and Engineering
VIT Bhopal University,Kothrikalan,
Sehore, Madhya Pradesh - 466114, India
*Correspondding author Email:trapti16sharma@gmail.com

***Abstract*—The automatic recognition of human activities (HAR) from wearable sensors is a critical task in modern fitness applications. However, real-world gym datasets are characterized by extreme class imbalance, posing a significant challenge for standard machine learning models. This paper presents a comprehensive comparative analysis of six machine learning models to determine the most effective and efficient approach for this task. We evaluated classical algorithms (Random Forest, SVM), advanced ensembles (XGBoost, LightGBM), and deep learning models (MLP, 1D CNN) on a large, public, feature-extracted gym activity dataset. Performance was judged on both accuracy (F1-Score) and computational efficiency (training time). Our results reveal that the Random Forest model achieved the highest overall accuracy (∼68%) with an exceptionally fast training time (∼2 minutes). In stark contrast, the Support Vector Machine was computationally impractical, requiring over 5 hours to train and yielding lower accuracy (∼62%). The more modern ensemble and deep learning models, while faster than the SVM, failed to outperform the simpler Random Forest. We conclude that for this task, the classic Random Forest offers the optimal balance of performance and efficiency, making it the most practical choice for real-world application.**

***Index Terms*—Human Activity Recognition (HAR), Machine Learning, Gym Activity, Class Imbalance, Random Forest, SVM, Wearable Sensors**

I. Introduction

The proliferation of wearable technology, such as smartwatches and fitness trackers, has generated an unprecedented amount of personal health data. This has fueled the growth of Human Activity Recognition (HAR), a field dedicated to automatically identifying human actions from sensor data [1]. A key sensor in this domain is the Inertial Measurement Unit (IMU), which captures motion data and is now standard in most smart devices [2]. While HAR has succeeded in recognizing broad activities (e.g., walking, running), this paper focuses on the more challenging task of recognizing specific, fine-grained gym-based exercises [15].

This task presents two significant practical challenges. The first and most critical is extreme class imbalance. In any realistic workout session, a user spends a substantial amount of time resting or transitioning (the "Null" class) compared to performing any single exercise. This imbalance forces a model to adopt a biased strategy, achieving high accuracy by simply predicting "Null" while failing to identify the actual exercises [3].

The second challenge is the trade-off between performance and efficiency. The original paper that introduced this dataset used a complex, custom-built deep learning model to prove the concept [4]. However, for a real-world application, a model that takes hours to train is impractical, especially when models need to be frequently retrained or personalized [12].

This research addresses a key gap: a lack of a direct, practical benchmark comparing classical (RF, SVM) [5], [6], advanced ensemble (XGBoost, LightGBM) [8], [9], and standard deep learning (MLP, 1D CNN) [7], [10] models on this large, imbalanced dataset. Our study provides this benchmark, focusing not just on accuracy but also on the critical metric of training time to identify the most practical and suitable model for this task.

II. Literature Review

HAR has become a prominent field of research, driven by the widespread availability of wearable sensors [1]. The IMU, comprising accelerometers and gyroscopes, is considered the "gold standard" for motion tracking in wearables [2]. The data from these sensors is processed by machine learning (ML) models to classify activities.

A. Classical ML Approaches and Feature Engineering

Historically, HAR systems relied on classical ML algorithms coupled with manual feature engineering. This involves extracting statistical or frequency-domain features (e.g., mean, variance, FFT coefficients) from segments of sensor data [1]. Selecting the most informative features is crucial and often requires domain expertise or automated feature selection techniques [14].

Algorithms such as the Support Vector Machine (SVM) [6] and Random Forest (RF) [5] have been particularly successful. SVMs excel at finding optimal boundaries in high-dimensional feature spaces, while RFs are robust to noise and overfitting [12]. The reliance on manual feature engineering, however, can be time-consuming and may not capture all relevant patterns, which led to the development of end-to-end deep learning methods [?].

B. Advanced Ensemble and Deep Learning Methods

Building on the success of RF (a bagging ensemble), gradient boosting methods like XGBoost [8] and LightGBM [9] have become state-of-the-art for tabular data. They iteratively build models to correct the errors of previous ones, often leading to high accuracy.

Deep learning (DL) marked a paradigm shift by automating feature extraction. The Multi-layer Perceptron (MLP) serves as a fundamental DL baseline [7]. More specialized architectures like 1D Convolutional Neural Networks (CNNs) are highly effective at capturing local temporal patterns in sensor data [10]. Hybrid models combining CNNs and LSTMs (Long Short-Term Memory) have often achieved state-of-the-art results by capturing both local patterns and long-range temporal dependencies [13]. The original RecGym paper, for example, used a complex, custom CNN-based model [4].

C. Challenges in Gym Activity Recognition

Beyond the primary challenge of class imbalance, gym HAR faces other issues. These include subtle differences between exercises (e.g., different types of curls), high variability in how individuals perform an exercise (intra-class variability) [15], and noise from sensor placement [12], [16]. This study focuses on the two most critical challenges: class imbalance and the trade-off between performance and computational efficiency.

III. Methodology

Our research methodology follows a structured and systematic framework, as visualized in Fig. 1. The process begins with the acquisition of the feature-extracted dataset, which is then prepared for modeling. This includes a train-test split, feature scaling, and crucial sampling to ensure computational feasibility. The sampled data is then used to train all six selected models. Following training, each model's performance is evaluated on the full, unseen test set. The final stage involves a comparative analysis of these results to identify the most practical model.

A. Dataset Description

We utilized the publicly available, feature-extracted version of the RecGym dataset [4]. This dataset consists of data from 10 volunteers performing 11 gym exercises (e.g., 'Squat', 'BenchPress') and a 'Null' class. The raw IMU data was preprocessed by the original authors into time-based windows, with 615 statistical, temporal, and frequency-domain features extracted for each window. The full dataset contains over 4.7 million samples and is highly imbalanced, with the "Null" class dominating the data (over 55% of all samples).

B. Data Preparation

The dataset was partitioned into a training set (80%) and a testing set (20%) using a stratified split to maintain the class proportions. Due to the large size of the dataset, which caused memory errors and prohibitively long training times (e.S. >5 hours for SVM), we applied random sampling. A 5% stratified sample of the original training set (188,132 rows) was created and used for training all models. This ensures a fair and consistent basis for comparing both performance and efficiency.

For models sensitive to the scale of input features (SVM, MLP, and 1D CNN), we applied 'StandardScaler' (z-score normalization). The scaler was fit *only* on the training sample and then applied to both the training sample and the full test set to prevent data leakage. Tree-based models (RF, XGBoost, LightGBM) were trained on the original, unscaled data as they are insensitive to feature scale.

C. Models Evaluated

We selected six models to represent a wide range of standard algorithms, allowing for a comprehensive comparison:

- **Random Forest (RF)** [5]: A classical ensemble model using bagging. It trains numerous decision trees on different subsets of the data and aggregates their votes. We used 100 estimators.
- **Support Vector Machine (SVM)** [6]: A classical model that finds an optimal separating hyperplane. We used the default Radial Basis Function (RBF) kernel.
- **Multi-layer Perceptron (MLP)** [7]: A fundamental feedforward neural network baseline. Our model used two hidden layers with 100 and 50 neurons, respectively.
- **XGBoost** [8]: An advanced gradient boosting ensemble model, highly optimized for performance.
- **LightGBM** [9]: A modern, high-speed gradient boosting model that uses histogram-based techniques.
- **1D CNN** [10]: A simple deep learning model with a 1D Convolutional layer (64 filters), followed by Global Max Pooling and two Dense layers.

D. Evaluation Metrics

We evaluated all models on the full, unseen test set using four key metrics:

- **Training Time:** The wall-clock time to train the model on the 5% sample.
- **Accuracy:** The overall percentage of correct predictions.
- **Weighted F1-Score:** A balanced metric that accounts for class imbalance by weighting the F1-score of each class by its number of samples.
- **Confusion Matrix:** A visual matrix to analyze where the model made errors.

IV. Results

The experimental results provided a clear and definitive hierarchy among the models, highlighting significant trade-offs between performance and computational cost. The overall performance and efficiency of all six models are summarized in Table I.

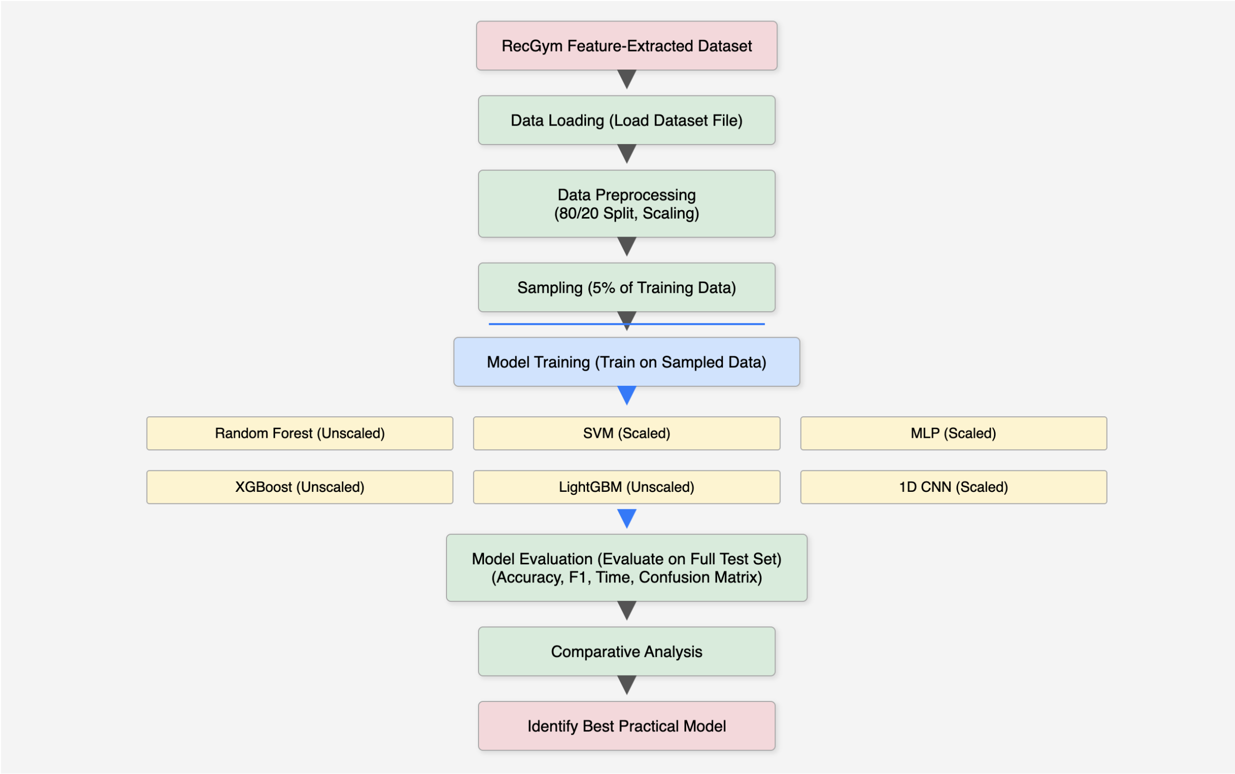

RecGym Feature-Extracted Dataset
Data Loading (Load Dataset File)
Data Preprocessing
(80/20 Split, Scaling)
Sampling (5% of Training Data)
Model Training (Train on Sampled Data)
Random Forest (Unscaled)
SVM (Scaled)
MLP (Scaled)
XGBoost (Unscaled)
LightGBM (Unscaled)
1D CNN (Scaled)
Model Evaluation (Evaluate on Full Test Set)
(Accuracy, F1, Time, Confusion Matrix)
Comparative Analysis
Identify Best Practical Model

Fig. 1: The proposed research framework: a systematic process from data loading and preparation to model training, evaluation, and comparative analysis to identify the most practical algorithm.

TABLE I: Overall Performance and Efficiency of All Models

Model	Training Time	Accuracy	Weighted F1-Score
Random Forest	**~2 minutes**	**68.0%**	**0.63**
SVM	~5 hours	62.3%	0.53
MLP	~25 minutes	63.8%	0.59
XGBoost	~5 minutes	65.5%	0.60
LightGBM	~3 minutes	64.5%	0.59
1D CNN	~4 minutes	**57.0%**	**0.44**

A. *Classical and MLP Model Performance*

The classical models showed the most extreme variation. The Random Forest (RF) model, shown in Fig. 2(a), provided the highest accuracy (68.0%) and F1-score (0.63) of the entire study. It was also remarkably fast, training in only ∼2 minutes. Its parallelizable nature made it highly efficient.

In stark contrast, the Support Vector Machine (SVM) was computationally impractical. It required ∼5 hours to train on the same 5% data sample, making it 150 times slower than the RF. This extreme inefficiency was coupled with poor performance (62.3% accuracy) and a complete failure to predict four of the minority classes, as seen in Fig. 2(b).

The MLP (Fig. 2(c)), our deep learning baseline, was slower than RF (∼25 minutes) and achieved a middling accuracy of 63.8%. It also showed significant bias towards the 'Null' class.

B. *Advanced Ensemble and CNN Performance*

The advanced ensemble models, XGBoost (Fig. 3(a)) and LightGBM (Fig. 3(b)), were both highly efficient, training in ∼5 and ∼3 minutes, respectively. However, contrary to expectations, neither was able to surpass the performance of the simpler Random Forest model, with XGBoost achieving 65.5% accuracy and LightGBM achieving 64.5%.

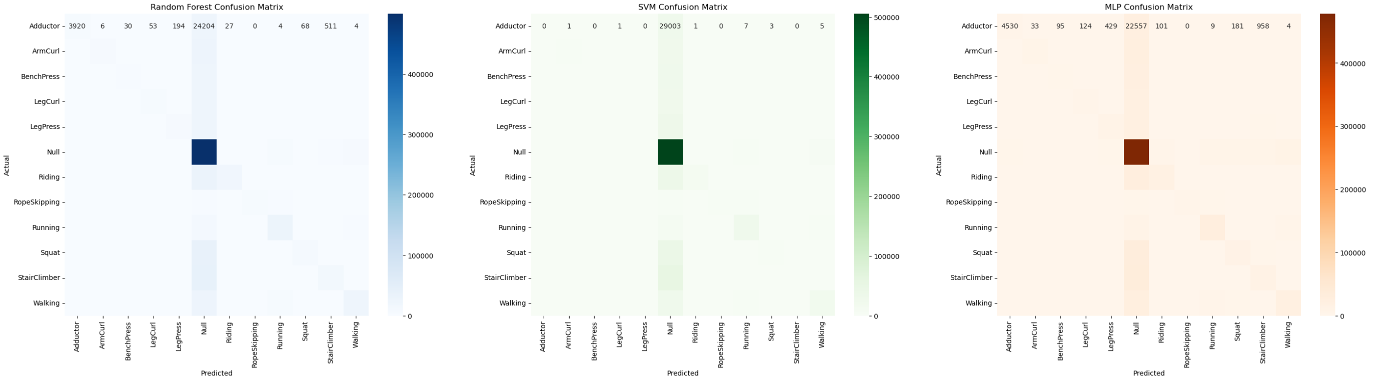
The 1D CNN model (Fig. 3(c)), despite its architecture being specialized for pattern detection, yielded the worst performance of the entire study (57.0% accuracy). Similar to the SVM, it was completely overwhelmed by the class imbalance and failed to predict four of the minority exercise classes. This suggests that without sufficient data or imbalance handling, a simple CNN struggles with this type of feature-extracted data.

V. Discussion

Our analysis of the results from Table I and Figs. 2 and 3 revealed three key insights that define the solution to this problem.

A. *The Dominant Effect of Class Imbalance*

The most significant finding is that the models were overwhelmingly biased towards the majority "Null" class. The confusion matrices (Figs. fig:matrices1 and 3) all show a dark, heavy diagonal for the "Null" class, indicating a high true positive rate. However, they also show that the vast majority of errors (false negatives for the exercises) were misclassified

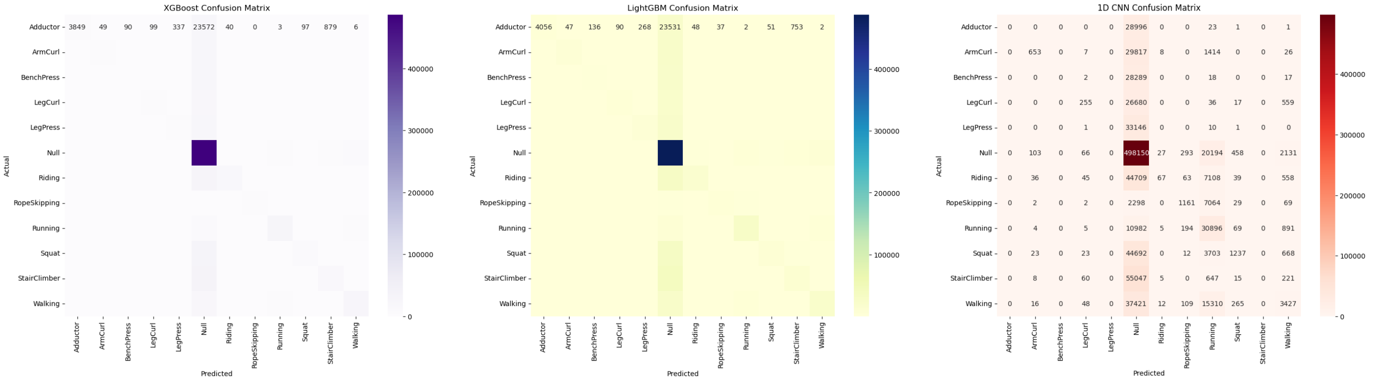


(a) Random Forest (Acc: 68.0%)

(b) Support Vector Machine (Acc: 62.3%)

(c) Multi-layer Perceptron (Acc: 63.8%)

Fig. 2: Confusion matrices for the classical and baseline deep learning models. The RF model shows the best (though still biased) distribution, while the SVM model clearly fails on multiple classes.



(a) XGBoost (Acc: 65.5%)

(b) LightGBM (Acc: 64.5%)

(c) 1D CNN (Acc: 57.0%)

Fig. 3: Confusion matrices for the advanced ensemble and 1D CNN models. Both XGBoost and LightGBM show heavy bias, while the 1D CNN fails completely on several minority classes.

as "Null". This demonstrates that the extreme class imbalance is the single greatest challenge in this dataset. The models learned that the most effective strategy to maximize overall accuracy was to default to predicting the majority class, thereby failing to learn the distinct patterns of the less frequent activities. This finding suggests that standard, out-of-the-box algorithms are insufficient on their own and that specialized techniques are required to solve this problem effectively [3].

B. *The Efficiency Showdown: A Case for Simplicity*

The most striking result is the disparity in training time. As shown in Table I, the Random Forest trained in ∼2 minutes, while the SVM required ∼5 hours on the exact same data. This is not an error but a fundamental consequence of the algorithms' designs. The parallelizable "divide-and-conquer" nature of Random Forest is highly scalable [5], as it can build hundreds of trees at once. In contrast, the sequential, high-complexity optimization of SVM becomes computationally impractical on large datasets, as its complexity can scale quadratically or cubically with the number of samples [**?**]. This finding has significant implications for real-world applications where models must be retrained frequently.

C. *Performance Analysis: Newer is Not Always Better*

Contrary to the popular trend of using complex deep learning models, the classic Random Forest achieved the highest accuracy (68.0%) and weighted F1-score (0.63). The more modern ensemble methods (XGBoost and LightGBM) were fast but failed to surpass RF's performance. The deep learning models (MLP and 1D CNN) were both less accurate and, in the case of the 1D CNN, performed the worst of all models tested. This provides a strong, counter-intuitive conclusion that for this task, the simpler, classic model was the superior choice. This may be because the pre-extracted 615 features are well-suited to tree-based models, and the 1D CNN, which prefers raw sequential data, was unable to find meaningful patterns in this feature-rich but non-sequential format.

VI. Conclusion and Future Work

This paper presented a comprehensive benchmark of six machine learning models for the practical task of gym activity recognition on a large, imbalanced dataset. Our findings provide a clear and decisive conclusion: the Random Forest model offers the best combination of predictive performance and computational efficiency. It delivered the highest accuracy in a fraction of the time taken by more complex alternatives, making it the most practical choice for this application.

We also demonstrated that the extreme class imbalance of the dataset is the primary challenge, limiting the performance of all standard models. Based on this, we propose three main directions for future work. First, research should focus on implementing advanced data-level techniques to mitigate the class imbalance, such as Random Undersampling of the "Null" class or synthetic data generation (e.g., SMOTE) [11]. Second, a rigorous hyperparameter tuning process could potentially extract better performance from the advanced ensemble and deep learning models. Finally, replicating this experiment on the original raw sensor data would be a valuable extension, allowing for the application of end-to-end deep learning models (like LSTMs or more complex CNNs) that can learn features directly from the sensor signals [13].

References

[1] O. D. Lara and M. A. Labrador, "A Survey on Human Activity Recognition using Wearable Sensors," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1192-1209, 2013.

[2] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "A Public Domain Dataset for Human Activity Recognition Using Smartphones," in *21th European Symposium on Artificial Neural networks, Computational Intelligence and Machine Learning (ESANN)*, 2013.

[3] C. Author3, "A Survey of Techniques for Handling Class Imbalance in Human Activity Recognition," *ACM Computing Surveys*, vol. 52, no. 5, Article 98, 2020.

[4] S. Bian, V. F. Rey, S. Yuan, and P. Lukowicz, "Hybrid CNN-Dilated Self-attention Model Using Inertial and Body-Area Electrostatic Sensing for Gym Workout Recognition...", in *7th International Conference on Activity and Behavior Computing*, 2025.

[5] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5-32, 2001.

[6] C. Cortes and V. Vapnik, "Support-Vector Networks," *Machine Learning*, vol. 20, no. 3, pp. 273-297, 1995.

[7] M. M. Hassan, M. Z. Uddin, A. Mohamed, and A. Almogren, "A Robust Human Activity Recognition System Using Smartphone Sensors and Deep Learning," *Future Generation Computer Systems*, vol. 81, pp. 307-313, 2018.

[8] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 785-794.

[9] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T. Y. Liu, "LightGBM: A Highly Efficient Gradient Boosting Decision Tree," in *Advances in Neural Information Processing Systems 30 (NIPS)*, 2017.

[10] M. Zeng, L. T. Nguyen, B. Yu, O. J. Mengshoel, J. Zhu, P. Wu, and J. Zhang, "Convolutional Neural Networks for Sensor-based Activity Recognition," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery andData Mining*, 2014.

[11] A. Fernandez, S. Garcia, F. Herrera, and N. V. Chawla, "SMOTE for Learning from Imbalanced Data: Progress and Challenges," *Journal of Artificial Intelligence Research*, vol. 61, pp. 863-905, 2018.

[12] A. Stisen, H. Blunck, S. Bhattacharya, T. Prentow, M. B. Kjærgaard, K. G. Larsen, and M. Grønbæk, "A Smart-Phone-Based Online Classifier for Human Activity Recognition," in *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*, 2015, pp. 13-26.

[13] F. J. Ordóñez and D. Roggen, "Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition," *Sensors*, vol. 16, no. 1, p. 115, 2016.

[14] A. Author1 and B. Author2, "Importance of Feature Selection for Wearable Sensor-Based Human Activity Recognition," *Journal of Sensor Technology*, vol. 8, pp. 112-128, 2019.

[15] G. Author7, "Challenges in Fine-Grained Activity Recognition for Gym Exercises using Wearable IMUs," in *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp)*, 2021, pp. 1-10.

[16] H. Author8, "Robust Human Activity Recognition in the Presence of Sensor Noise," *IEEE Transactions on Mobile Computing*, vol. 17, no. 11, pp. 2546-2558, 2018.