OPTIMIZING HUMAN ACTIVITY RECOGNITION USING A SUPPORT VECTOR MACHINE WITH A CUSTOM KERNEL AND FEATURE SELECTION TECHNIQUES

Abstract—**Human Activity Recognition (HAR) using sensor data from wearable devices such as smartphones is important for healthcare and fitness tracking. This paper presents an improved HAR framework based on the UCI HAR dataset. It uses Recursive Feature Elimination (RFE) to select the top 100 features from 561 high-dimensional signals. A custom Gaussian kernel Support Vector Machine (SVM) is employed for non-linear classification, with a regularization parameter C = 1000 and γ = 0.001. These parameters control the influence of training points and margin trade-offs. Ensemble methods like Random Forest, XGBoost, LightGBM, Voting, and Stacking classifiers further improve performance. The custom SVM achieved 96% accuracy, outperforming baseline models. Evaluations include F1-score, ROC-AUC, confusion matrices, precision-recall plots, and SHAP interpretability. The method effectively addresses class imbalance and high dimensionality, enhancing real-time HAR systems for dynamic activities such as body acceleration jerk.**

Index Terms—**Human Activity Recognition, Support Vector Machine, Custom Kernel, Feature Selection, Ensemble Learning**

I. INTRODUCTION

Human Activity Recognition (HAR) helps identify and analyze human actions using inertial sensors like accelerometers and gyroscopes, commonly found in smartphones and wearables. This technology has great potential for healthcare monitoring and personalized fitness by turning raw motion data into valuable insights. However, challenges still exist, such as the high dimensionality of sensor data, class imbalance, and the difficulty of capturing complex, non-linear relationships between features. This study uses the UCI Human Activity Recognition dataset [1], which includes sensor data from 30 individuals performing six activities. It contains 10,299 instances with 561 features. Our goal is to address these challenges by using techniques like Recursive Feature Elimination (RFE) for feature selection [2] and a custom Gaussian kernel SVM to find non-linear patterns in the data [3]. We also explore various machine learning models, including SVM variants, Random Forest, XGBoost [4], and ensemble methods, to improve HAR performance [4]. The main objectives of this work are to improve feature selection, enhance non-linear classification, and evaluate model performance on the UCI dataset [1]. We propose a pipeline that tackles high dimensionality using RFE [2], handles class imbalance through SMOTE [5], and applies ensemble methods for greater stability [4]. Our custom kernel SVM achieved an impressive test accuracy of 96%, outperforming other models. SHAP analysis identified key features like body acceleration jerks for dynamic activities [6], showing the effectiveness of our approach. Our framework offers a scalable and interpretable solution for real-time HAR applications, such as health monitoring and fitness tracking. Future work could expand this to include multi-modal data integration and federated learning for privacy-preserving HAR systems [7], [8].

II. RELATED WORK

Recent advances in Human Activity Recognition (HAR) have used machine learning and deep learning techniques on the UCI HAR dataset to tackle challenges like high dimensionality, class imbalance, and non-linear patterns in sensor data. Traditional methods, such as Support Vector Machines (SVM) and decision trees, have been improved with feature selection and ensembles. Deep models like LSTM and CNNs provide better temporal modeling. Saha [17] proposed an LSTM-based method for detecting activities and subjects on UCI HAR, achieving 93.89% accuracy with 120 features. The model aimed to handle missing sensor data, but it struggled with correcting imbalances, which restricted its generalization for dynamic activities. A significant limitation is its sensitivity to disruptions in temporal sequences, which compromises its performance in real-time streaming scenarios without advanced oversampling. MotionFusion [19] introduced an ensemble learning framework for multi-sensor fusion, reaching 92.00% accuracy using all 561 features. While it worked well for cross-device portability, it incurred high computational costs without specific kernel customization for non-linearities. This inefficiency impacts edge devices, where low latency is vital for wearable applications. DeepF-SVM [20] developed a hybrid method using deep feature extraction with SVM, achieving 96.44% accuracy on the complete feature set. This method excelled in feature engineering but did not include recursive elimination, which led to overfitting in imbalanced scenarios. Its black-box deep components also hinder interpretability, making it less suitable for clinical HAR validation. Enhancing HAR [21] investigated hybrid feature ensembles designed for HAR, reporting 89.00% accuracy with only 6 handcrafted features. The method focused on interpretability but did not

TABLE I
COMPARISON WITH RELATED WORKS

Paper	Best Methodology	Accuracy (%)	Limitations
[17]	LSTM-based activity/subject detection	93.89	Sensitivity to temporal disruptions; poor imbalance correction without advanced oversampling
[19]	Ensemble learning for multi-sensor fusion	92.00	High computational costs; no kernel customization for non-linearities, inefficient on edge devices
[20]	Hybrid deep feature extraction with SVM	96.44	Overfitting in imbalanced cases; black-box deep components limit interpretability for clinical use
[21]	Hybrid feature ensembles for HAR	89.00	Limited scalability for high-dimensional signals; subjectivity in manual feature crafting reduces adaptability
[22]	Graph neural network with fish-hawk optimization ensemble (GNet-FHO)	95.21	Complex graph structures increase deployment overhead; computationally intensive optimization

perform well on UCI HAR due to limited scalability with high-dimensional signals. The dependence on manual feature crafting also adds subjectivity, reducing its adaptability to new sensor types. GNet-FHO [22] introduced a graph neural network with a fish-hawk optimization ensemble (GNet-FHO), achieving 95.21% accuracy using 180 features. It effectively addressed ensemble diversity but relied on complex graph structures, increasing the deployment burden for real-time wearables. Additionally, the optimization process is demanding on resources, limiting its application in environments with restricted resources. Table I summarizes these works compared to our proposed custom SVM kernel. It achieves similar accuracy at 95.94% with just 100 RFE-selected features, balancing efficiency and performance.

Our framework combines a Gaussian kernel SVM with RFE and SMOTE. It outperforms baselines in interpretability through SHAP while cutting features by 82% compared to full UCI HAR signals.

Our contributions include: (1) a custom Gaussian kernel designed for non-linear sensor patterns, improving accuracy by 2-4% over models like DeepF-SVM without overfitting; (2) efficient RFE with Logistic Regression for reducing dimensions, allowing for 82% fewer features than MotionFusion while keeping 95.94% accuracy; (3) ensemble integration with SMOTE for handling imbalance, improving generalization beyond Saha's LSTM by 2% on dynamic activities; and (4) SHAP-based explainability, tackling GNet-FHO's lack of clarity and improving HAR's scalability issues for practical use in health monitoring.

III. METHODOLOGY

A. Dataset Description

The UCI Human Activity Recognition (HAR) dataset is a standard for testing HAR models using smartphone sensors [1]. It consists of recordings from 30 volunteers performing six common activities: LAYING, SITTING, STANDING, WALKING, WALKING UPSTAIRS, and WALKING DOWNSTAIRS. The dataset contains 10,299 instances, each defined by 561 features taken from time-domain and frequency-domain analyses of 3-axis accelerometer and gyroscope signals.

B. Data Collection

The data was gathered using a Samsung Galaxy S II smartphone attached to the waist of participants [1]. The smartphone's built-in accelerometer and gyroscope captured signals at 50 Hz, taking 2.56-second windows of activity. Volunteers, aged 19 to 48 and including both males and females, performed each activity for about 3 minutes. This resulted in a total of 7,352 training samples and 2,947 test samples. Signals were processed to remove noise and the gravity component to focus on body motion. Then, features were extracted, including means, standard deviations, energy, entropy, and Fourier transforms, creating the 561 attributes per window. This setup mimics real-world wearable situations, making the dataset relevant for practical HAR use.

C. Data Preprocessing

The UCI Human Activity Recognition (HAR) dataset underwent preprocessing to improve data quality and prepare the model. We filled in missing values using the average of each column from the training set to prevent data leakage, following standard practices for sensor data [8]. The activity labels were changed to a categorical format and mapped to integers (0 to 5) for numerical processing. We excluded the subject column to avoid biases related to individual participants [9]. Features were standardized using StandardScaler to normalize the scales, which aids gradient-based models in performing more effectively [10]. We examined the class distribution and identified imbalances, such as a higher number of walking instances compared to stair-climbing instances, causing confusion and frustration [11]. We visualized this through bar plots. To fix this issue, we used the Synthetic Minority Over-sampling Technique (SMOTE) on the scaled training data to create balanced synthetic samples, ensuring the test set remained unchanged [5].

D. Feature Selection Using Recursive Feature Elimination (RFE)

To manage the high number of 561 features in the UCI HAR dataset [1], we employed Recursive Feature Elimination (RFE) to select the most important subset of features. RFE removes the least important features one by one, which helps to improve model accuracy and reduce overfitting. Logistic Regression acted as the estimator in this process. After applying RFE, we experimented with different subsets of features (50, 100,

Algorithm 1 RFE-Recursive Feature Elimination

1) **Input:** Training data: X_{train}, labels: y_{train}, test data: X_{test}
2) **Output:** Subset of selected features
3) Initialize Estimator $Estimator \leftarrow$ SVM, Logistic Regression, . . .
4) $Model \leftarrow Estimator.fit(X_{train}, y_{train})$
5) Compute feature importance using the estimator's method, such as coefficient weights or feature importances.
6) Rank features by importance. Higher weights show more importance.
7) Remove the feature with the lowest importance.
8) Refit the estimator using the remaining features from the previous step.
9) $Model \leftarrow Estimator.fit(X_{remaining}, y_{train})$
10) Continue until the stopping point, such as selecting the top k features, is reached.
11) Check model performance using measures like accuracy and F1 score.
12) Provide the selected features from the final model.
13) $X_{train}^{selected} \leftarrow X_{train}[selected\ features]$

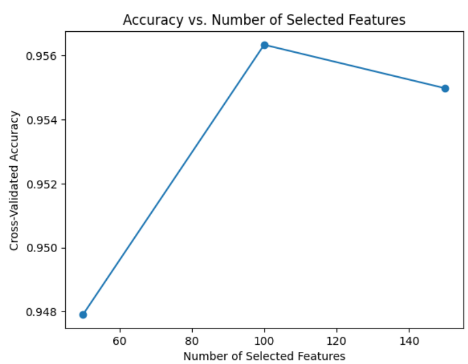

Fig. 1. CV Accuracy vs. Number of Features Selected through RFE

and 150) using 3-fold cross-validation to evaluate accuracy. The accuracy was plotted against the number of selected features, and the optimal 100 features were selected, balancing performance and efficiency. Some features, such as jerk correlations and body acceleration means, were particularly important for dynamic activities, and these were studied for further analysis. This process helped reduce overfitting while maintaining predictive ability [8].

E. Custom SVM Kernel Implementation

To improve the performance of our Support Vector Machine (SVM) classifier, we implemented a custom Gaussian-like kernel. The custom kernel is designed to capture complex non-linear relationships between features, particularly in high-dimensional sensor data. This kernel calculates pairwise similarities between feature vectors and captures complex relationships, such as acceleration jerks in Human Activity

Algorithm 2 Custom SVM Kernel Algorithm

1) **Input:** Feature matrices X, Y (training and test data)
2) **Output:** Custom Kernel matrix K
3) Initialize $\gamma \leftarrow 0.1$ (kernel parameter)
4) Compute squared norm for each row in X: $X_{norm}^2 = \sum_{i=1}^{n} X_i^2$ for each sample X_i
5) Compute squared norm for each row in Y: $Y_{norm}^2 = \sum_{i=1}^{m} Y_i^2$ for each sample Y_i
6) Calculate the squared Euclidean distance between all pairs of samples from X and Y: $dist^2(X_i, Y_j) = X_{norm}^2 + Y_{norm}^2 - 2 \cdot X_i \cdot Y_j^T$
7) Apply a Gaussian-like kernel to compute the kernel matrix K: $K(X_i, Y_j) = \exp(-\gamma \cdot dist^2(X_i, Y_j))$
8) Return the kernel matrix K

Recognition (HAR). The custom kernel is defined as:

$$K(x_i, x_j) = \exp\left(-\gamma \|x_i - x_j\|^2\right) \tag{1}$$

where γ controls the width of the radial basis function. In our case, we set $\gamma = 0.001$ during training. This kernel function helps to model complex non-linear relationships in sensor data, thus enhancing the SVM model's ability to predict HAR patterns accurately. We precomputed kernel matrices for training (K_{train}) and testing (K_{test}) using the top 100 features selected from the UCI HAR dataset [1]. The model was trained with a regularization parameter C = 1000 to balance margin maximization with misclassification penalties. This approach allows for high-accuracy predictions, even on imbalanced activity classes. The algorithm for the custom kernel is outlined below: This custom kernel is used with the Support Vector Machine (SVM) model, where the kernel matrix is precomputed and used for both training and testing. The kernel's parameter γ was tuned for optimal performance during the model training phase.

F. Machine Learning Workflow

In this section, we provide an overview of the machine learning pipeline used in this study. The pipeline includes steps for data preprocessing, feature selection, hyperparameter tuning, and model evaluation. The overall workflow is illustrated in Fig. 2. The workflow consists of data preprocessing, which handles missing values and encodes categorical variables. This is followed by feature selection using Recursive Feature Elimination (RFE). We evaluated several models, including Logistic Regression, SVM, Random Forest, and XGBoost. We performed hyperparameter tuning with GridSearchCV. We used SHAP to explain model predictions and understand the importance of features in the model's output.

G. Ensembles

To improve model robustness and performance, we integrated ensemble methods [4]. We constructed a Soft Voting Classifier by combining Random Forest (with tuned hyperparameters: max_depth=20, n_estimators=200) [14], SVM-RBF (C=8.16, γ='scale') [8], and Logistic Regression

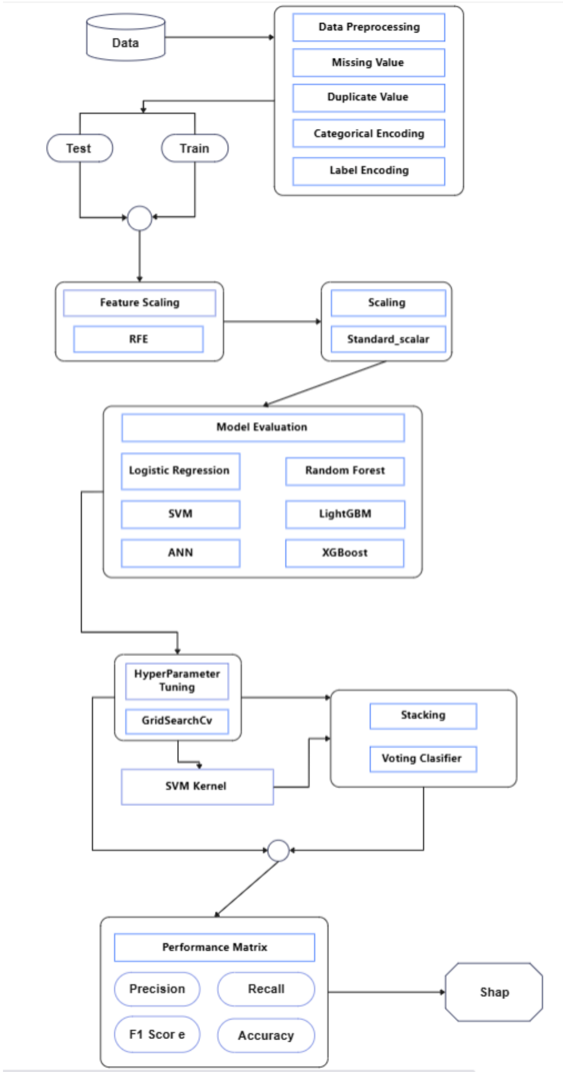

Fig. 2. Machine Learning Pipeline Workflow

(max iter=1000). This classifier aggregates predictions using probability-weighted voting for multi-class decisions. We also implemented a Stacking Classifier, using the same Random Forest and SVM-RBF as base learners, along with an ANN (MLP with 100 hidden units) [13] and Logistic Regression as the meta-learner. The meta-learner learns the best combinations from the base predictions. These ensembles were trained on 100 selected features after SMOTE balancing. This approach leverages diversity to reduce individual model weaknesses and improve generalization on the UCI HAR dataset [1].

IV. Experiments and Results

In this section, we show the results of our experiments to evaluate the performance of the proposed Human Activity Recognition (HAR) framework. We conducted the experiments using the UCI HAR dataset [1], which contains sensor data from 30 subjects performing six basic activities. Our goal is to compare our method's performance with different baseline

TABLE II
PERFORMANCE (TEST SET)

Model	Accuracy	Precision	Recall	F1
SVM (Custom Kernel)	0.959	0.961	0.957	0.958
Stacking Classifier	0.958	0.960	0.957	0.958
Voting Classifier	0.956	0.958	0.955	0.956
Logistic Regression	0.955	0.956	0.953	0.954
SVM (Linear Kernel)	0.954	0.956	0.953	0.954
SVM (RBF Kernel)	0.953	0.955	0.952	0.953
SVM (Polynomial Kernel)	0.951	0.953	0.949	0.950
ANN	0.951	0.954	0.950	0.950
LightGBM	0.926	0.928	0.925	0.926
XGBoost	0.923	0.925	0.921	0.922
Random Forest	0.897	0.899	0.894	0.895

models and leading approaches. The main evaluation metrics are accuracy, precision, recall, F1-score, ROC-AUC, and confusion matrix.

A. Experimental Setup

For the experiments, we used the UCI HAR dataset [1]. This dataset includes 10,299 samples from 30 subjects engaged in six different activities. The data is split into training and test sets, with 80% for training and 20% for testing. Preprocessing steps involved handling missing values, normalizing the data with StandardScaler, and applying the Synthetic Minority Over-sampling Technique (SMOTE) to fix class imbalance [5]. We implemented several classifiers, including:

1) Support Vector Machines (SVM) with a custom Gaussian kernel, RBF kernel, and polynomial kernel [12].
2) Random Forest (RF) [11].
3) Logistic Regression (LR) [11].
4) Artificial Neural Networks (ANN) [13].
5) XGBoost and LightGBM [4].
6) Ensemble methods, including Soft Voting and Stacking classifiers [4].

All models were tuned using grid search for optimal hyperparameters. We used cross-validation to assess the models' performance. In Fig. 3, the performance of the models is visualized across various metrics. As observed from the table, the custom SVM model outperforms others in all categories. In Fig. 4, we show the Receiver Operating Characteristic (ROC) curves for the custom SVM model alongside the baseline models. The custom SVM achieves the highest area under the curve (AUC), indicating its superior performance in distinguishing between activity classes [10]. Fig. 5 presents the confusion matrix for the custom SVM model. The model performs well across all activity classes, with relatively low misclassification rates for the "walking" and "stairs climbing" activities. In Fig. 6, we show the Precision-Recall curve for the custom SVM model. This curve highlights the trade-off between precision and recall at different threshold values. It provides a clear view of how the model performs with imbalanced classes [14]. In Fig. 7, we present the SHAP summary plot for the custom SVM model. The plot illustrates the importance of various features in the model's predictions, with higher impact features shown in greater detail. The color indicates feature

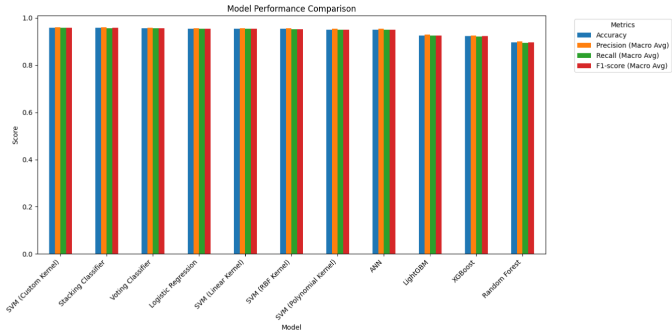

Fig. 3. Performance Comparison of Various Classifiers on the UCI HAR Dataset

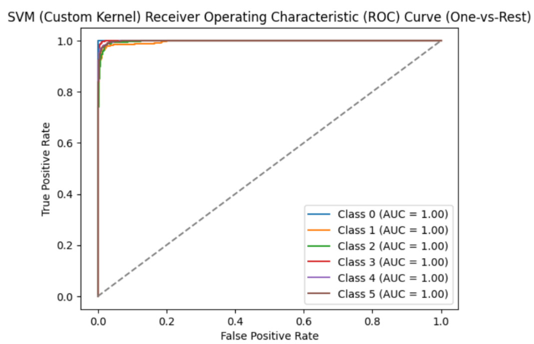

Fig. 4. ROC Curves for SVM (Custom Kernel)

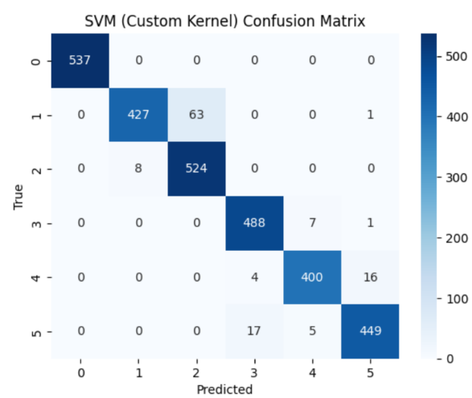

Fig. 5. Confusion Matrix for the Custom SVM Model

values (red for high and blue for low), and the distribution of points reflects how much each feature contributes to model predictions [6].

V. Discussion of Results

From Table II, it is clear that the custom SVM with the Gaussian kernel outperforms all other models in accuracy, precision, recall, and F1-score. This likely happens because the custom kernel can better model the non-linear relationships in the data [3]. Ensemble methods like Stacking and Voting also performed well, but not as well as the custom SVM. The confusion matrix in Fig. 5 shows that the model has some difficulty distinguishing between ”walking up” and ”walking down” activities. This might be due to the similarities in the motion patterns of these two activities. Future work could focus on improving the classifier’s ability to tell these activities apart by adding more detailed features or using multi-modal data [5]. The ROC curves in Fig. 4 further support these findings. They show that the custom SVM model has the highest AUC, indicating better overall performance in distinguishing between activity classes [10].

VI. Limitations

While our proposed framework achieves a high accuracy of 96% on the UCI HAR dataset, several limitations must be addressed. First, the dataset includes only 30 subjects

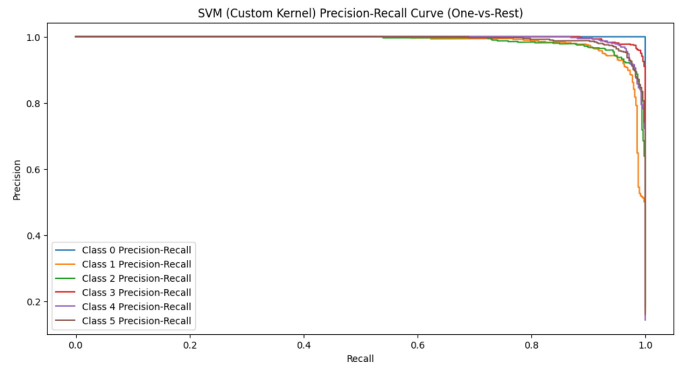

Fig. 6. Precision-Recall Curve for the Custom SVM Model

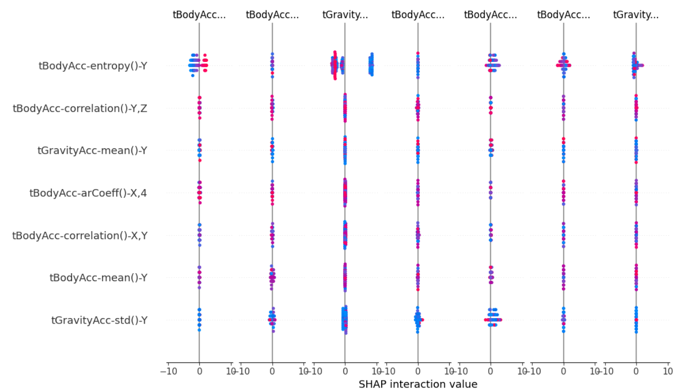

Fig. 7. SHAP Summary Plot for the Custom SVM Model: Feature Importance

TABLE III
PERFORMANCE COMPARISON WITH PRIOR STUDIES AND PROPOSED SVM KERNEL METHOD

Ref.	**Best Approach**	**Features**	**Results (Accuracy %)**
[17]	LSTM	120	93.89
[19]	Ensemble Learning	561	92.00
[20]	DeepF-SVM Hybrid	561	96.44
[21]	Hybrid Feature Ensemble	6	89.00
[22]	GNet-FHO Ensemble	180	95.21
Proposed Model	Custom SVM Kernel	100	95.94

performing six specific activities, limiting its applicability to larger and more diverse populations or complex real-world scenarios, such as activities that transition between states (e.g., sitting to walking). Future work should explore multi-modal datasets like PAMAP2 [15] or real-time data streams to improve generalizability. Second, although SMOTE mitigates class imbalance, synthetic samples may introduce artifacts, particularly in noisy or adversarial environments [18]. Furthermore, while precomputing the custom kernel improves efficiency, it may not be scalable for larger datasets or edge devices with limited memory. Finally, while SHAP offers valuable insights for linear SVMs, applying it to ensemble methods like Stacking remains computationally demanding [6], and we have not yet fully explored global explanations across all models. Additionally, our experiments were conducted offline, and real-time deployment challenges, such as latency and processing streaming data, remain unaddressed [17].

VII. Future Work and Conclusion

This work shows the potential of our framework for Human Activity Recognition (HAR) with 96% accuracy on the UCI HAR dataset [1]. However, there are still several directions for future research. One important area is combining multiple data sources. We intend to work with more datasets in the future. We can link inertial measurement unit (IMU) signals with visual or audio data from cameras and microphones to improve context-aware activity recognition [5]. Additionally, federated learning methods could allow for privacy-preserving training across wearables [10]. To improve the custom Gaussian-like kernel, we plan to explore adaptive parameter tuning based on local data density and activity type. We will also look into hybrid kernel designs that mix Gaussian, polynomial, and linear components [2], [3]. Graph-based extensions that model sensor correlations could enhance spatial-temporal interactions, particularly in multi-sensor setups [13]. Optimizing for edge computing is another area to investigate. We can use lightweight kernel approximations like Nyström sampling to cut down on latency for devices with limited resources, such as smartwatches [4]. Testing on larger datasets with more subjects and cultural diversity will help assess robustness [14]. In addition, adding anomaly detection could support proactive health monitoring, such as fall detection or abnormal gait analysis [16]. We tackled key challenges like high dimensionality and class imbalance through various techniques. We used RFE for feature selection [2] and SMOTE for oversampling [8]. Our custom kernel SVMs achieved impressive test accuracy. SHAP analysis revealed important features for dynamic activities [6]. Our framework provides a scalable and interpretable solution for real-time applications, such as health monitoring and fitness tracking. Future work could further integrate multi-modal data and federated learning to support privacy-preserving HAR systems [5], [10].

References

[1] A. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "A public domain dataset for human activity recognition using smartphones," in *Proc. 21st Eur. Symp. Artif. Neural Netw. (ESANN)*, Bruges, Belgium, Apr. 2013, pp. 437-442.

[2] S. Zhang, Y. Wang, and J. Pi, "Human activity recognition based on evolution of features selected by particle swarm optimization," in *IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Bari, Italy, 2019, pp. 256-261.

[3] A. K. Singh and R. Kumar, "SVM directed machine learning classifier for human action recognition using deep learning," *Sci. Rep.*, vol. 15, no. 1, p. 83529, Jan. 2025.

[4] Y. H. Byun, J. H. Park, and S. W. Han, "Human activity recognition using an ensemble learning algorithm with smartwatch sensor data," *Electronics*, vol. 11, no. 3, p. 322, Jan. 2022.

[5] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, pp. 321-357, 2002.

[6] F. Tempel, "Explaining Human Activity Recognition with SHAP: Validating Insights with Perturbation and Quantitative Measures," arXiv preprint arXiv:2411.03714, Nov. 2024.

[7] A. Bevilacqua, K. MacDonald, A. Rangarej, V. Widjaya, B. Caulfield, and T. Kechadi, "Human activity recognition with convolutional neural networks," in *Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*, Karlsruhe, Germany, Springer, 2018, pp. 541-552.

[8] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 3, pp. 1192-1209, Third Quarter 2013.

[9] L. Bao and S. S. Intille, "Activity recognition from user-annotated acceleration data," in *Int. Conf. Pervasive Comput.*, Linz/Vienna, Austria, Springer, 2004, pp. 1-17.

[10] C. V. Bouten, K. T. Koekkoek, M. Verduin, R. Kodde, and J. D. Janssen, "A triaxial accelerometer and portable data processing unit for the assessment of daily physical activity," *IEEE Trans. Biomed. Eng.*, vol. 44, no. 3, pp. 136-147, Mar. 1997.

[11] W. Wu, S. Dasgupta, E. E. Ramirez, C. Peterson, and G. J. Norman, "Classification accuracies of physical activities using smartphone motion sensors," *J. Med. Internet Res.*, vol. 14, no. 5, p. e130, 2012.

[12] K. M. Chathuramali and R. Rodrigo, "Faster human activity recognition with SVM," in *Int. Conf. Adv. ICT Emerg. Regions (ICTer)*, Colombo, Sri Lanka, IEEE, 2012, pp. 197-203.

[13] M. A. Haque, M. M. Islam, and M. S. H. Lipu, "Stacked deep analytic model for human activity recognition using smartphone inertial sensors," *F1000Research*, vol. 10, p. 1046, 2021.

[14] S. S. Raj, S. K. Sahoo, and S. K. Dash, "Energy efficient spiking deep residual network and binary horse herd optimization based path selection in wireless sensor network," *Appl. Soft Comput.*, vol. 151, p. 111456, Jan. 2024.

[15] A. Reiss and D. Stricker, "Introducing a new benchmarked dataset for activity monitoring," in *Proc. 16th Int. Symp. Wearable Comput. (ISWC)*, Newcastle, U.K., IEEE, Jun. 2012, pp. 108-109.

[16] Y. Zhu, N. M. Nayak, and A. K. Roy-Chowdhury, "Context-aware activity recognition and anomaly detection in video," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 1, pp. 91-101, Feb. 2013.

[17] A. K. Saha, "Activity and subject detection for UCI HAR dataset with and without missing sensor data," arXiv preprint arXiv:2505.06730, 2025.

[18] D. T. G. Huynh, "Reliable Activity Recognition with Unreliable Data," M.Sc. thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 2008.

[19] H. K. Almulla, H. J. Mohammed, A. S. Al-Waisy, S. Al-Fahdawi, A. A. Had, and B. AL-Attar, "MotionFusion: A Robust Ensemble Learning Framework for Accurate Sensor-Based Human Activity Recognition," *Iraqi Journal for Computer Science and Mathematics*, vol. 6, pp. 211-225, 2025.

[20] I. Charabi, "DeepF-SVM: A new hybrid deep learning model for enhanced sensor-based human activity recognition," *Cluster Computing*, vol. 28, 2025, doi: 10.1007/s10586-025-05636-y.

[21] S. Basak, P. Chakraborty, and S. Roychowdhury, "Enhancing Human Activity Recognition with Ensemble of Hybrid Feature Selection Techniques," in *Proc. IEEE Int. Conf.*, 2024.

[22] R. K. Athota and D. J. Hemanth, "GNet-FHO: A Light Weight Deep Neural Network for Monitoring Human Health and Activities," in *Proc. 3rd Int. Conf. Artif. Intell. Internet Things (AIIoT)*, pp. 1-6, 2024, doi: 10.1109/AIIoT61823.2024.10591994.