Black Sea Journal of Engineering and Science
doi: 10.34248/bsengineering.1997666

Open Access Journal
e-ISSN: 2619 – 8991

Research Article
Volume 9 - Issue 5: 2612-2620 / September 2026

COMPARATIVE ANALYSIS OF METAHEURISTIC FEATURE SELECTION IN SVM-BASED HUMAN ACTIVITY RECOGNITION

Duygu Sedef ÇALIŞKAN[1], Özlem KILIÇ[1], Durmuş Özkan ŞAHİN[1], Gökhan KAYHAN[1]*, Sercan DEMİRCİ[1]

[1]*Ondokuz Mayıs University, Department of Computer Engineering, 55139 Samsun, Türkiye*

Abstract: Smartphone-based human activity recognition (HAR) commonly relies on large sets of engineered time- and frequency-domain variables. Although these representations capture complementary motion patterns, redundant variables may increase model cost without resolving ambiguity between similar activities. This study compared a genetic algorithm (GA), particle swarm optimization (PSO), and grey wolf optimization (GWO) as wrapper-based feature selectors for a radial basis function support vector machine (SVM) on the UCI HAR dataset. The original 7,352/2,947 training-test split and a common SVM configuration were retained. Each optimizer searched a 561-bit feature mask with a population of 10 for 10 generations or iterations, using an objective that prioritized predictive accuracy while penalizing subset size. Performance was assessed by held-out accuracy, weighted F1, selected features, confusion patterns, convergence, and total runtime. The baseline SVM achieved 0.9308 accuracy and 0.9304 weighted F1 with all 561 features. GA+SVM produced the strongest result: 0.9508 accuracy, 0.9506 weighted F1, 300 features, and 217.99 s. This reduced dimensionality by 46.52% and test errors from 204 to 145. GWO+SVM reached 0.9471 accuracy with 341 features and 295.12 s. PSO+SVM selected the smallest subset (284 features; 49.38% reduction), but its 0.9355 accuracy and 489.68 s runtime yielded a weaker trade-off. Errors remained concentrated in upstairs/downstairs and sitting/standing/laying distinctions. Under the tested search budget, GA+SVM offered the best observed balance between predictive performance, compactness, and optimization cost. Because the stochastic algorithms were not evaluated over repeated independent runs, the ranking should be interpreted as configuration-specific rather than universal.

Keywords: Human activity recognition, Feature selection, Support vector machine, Genetic algorithm, Particle swarm optimization, Grey wolf optimization

***Corresponding author:** Ondokuz Mayıs University, Department of Computer Engineering, 55139 Samsun, Türkiye
E mail: gkayhan@omu.edu.tr (G. KAYHAN)

Duygu Sedef ÇALIŞKAN https://orcid.org/0009-0005-7588-3498
Özlem KILIÇ https://orcid.org/0009-0002-6780-4284
Durmuş Özkan ŞAHİN https://orcid.org/0000-0002-0831-7825
Gökhan KAYHAN https://orcid.org/0000-0003-3391-0097
Sercan DEMİRCİ https://orcid.org/0000-0001-6739-7653

Received: July 17, 2026
Accepted: August 19, 2026
Published: September 15, 2026

Cite as: Çalışkan, D. S., Kılıç, Ö., Şahin, D. Ö., Kayhan, G., & Demirci, S. (2026). Comparative analysis of metaheuristic feature selection in SVM-based human activity recognition. *Black Sea Journal of Engineering and Science,* 9(5), 2612-2620.

1. Introduction

Human activity recognition (HAR) seeks to infer everyday actions from signals recorded by phones, watches, and other wearable devices. Reliable recognition supports health monitoring, rehabilitation, assisted living, context-aware services, and safety applications. Early wearable computing studies established that recurring behavioral patterns could be recovered from body-worn sensors (Clarkson, 2002), while later work showed that ordinary phone accelerometers were sufficient to distinguish common activities in practical settings (Kwapisz et al., 2011). Recent HAR research has expanded to richer sensor combinations and deep models, yet the cost and interpretability of the input representation remain relevant for deployable systems (Vurgun and Kıran, 2024). Recent datasets are also broadening the evaluation landscape. For example, Tokgöz and Kocaoğlu (2026) introduced AybuFall, an openly available IMU dataset that combines falls, activities of daily living, and prayer movements, illustrating the continuing need for robust and transferable HAR evaluations.

The UCI Human Activity Recognition Using Smartphones dataset provides a controlled benchmark for studying this issue. It contains accelerometer and gyroscope measurements collected from 30 volunteers performing six activities, together with 561 engineered variables derived from fixed-width signal windows (Anguita et al., 2013). These variables summarize body acceleration, gravity, angular velocity, jerk, magnitude, and frequency-domain behavior. Such representations remain useful because they can be evaluated with conventional classifiers and modest hardware, an important consideration alongside the rapid growth of end-to-end smartphone HAR methods (Dentamaro et al., 2024).

A high-dimensional representation does not automatically yield a stronger classifier. Correlated, duplicated, or weakly discriminative variables can increase training and inference cost, make model selection less stable, and obscure which signal

characteristics carry useful information. Feature selection addresses this problem by retaining original variables while discarding those that contribute little to prediction. Filter, wrapper, and embedded methods differ in how subsets are evaluated, but all aim to construct a lower-dimensional and more informative representation (Tang et al., 2014; Budak, 2018).

Wrapper selection is attractive when the usefulness of a variable depends on the classifier and on interactions with other variables. Its main difficulty is computational: a dataset with 561 features defines 2^{561} possible subsets. Evolutionary and swarm-based algorithms provide a practical way to sample this search space without exhaustive enumeration. Reviews of metaheuristic feature selection show that these methods can improve classification while reducing dimensionality, but they also stress the need to report computational effort and to compare methods under common conditions (Xue et al., 2016; Agrawal et al., 2021; Akinola et al., 2022). Turkish-language overviews reach a similar conclusion: the optimizer should be selected according to the problem structure and evaluation budget rather than by its popularity alone (Gazioğlu, 2024; Tonguç, 2024).

The genetic algorithm (GA) explores binary subsets through selection, crossover, mutation, and elitist retention. Its population-based recombination can preserve useful feature groups while mutation introduces new subsets, which explains its long-standing use in feature selection (Hussein et al., 2001; Babatunde et al., 2014). Hybrid and task-specific GA designs have also shown that the balance between exploration and exploitation matters as much as the encoding itself (Oh et al., 2004).

Particle swarm optimization (PSO) instead updates candidate solutions from personal and collective best positions. Binary PSO variants translate this movement into inclusion decisions and have been applied to both single- and multi-objective feature selection (Chuang et al., 2008; Xue et al., 2013). Grey wolf optimization (GWO) organizes the population around three leading solutions and moves the remaining agents toward this leadership hierarchy. Binary and multi-objective GWO variants have produced compact subsets in classification tasks (Emary et al., 2016; Al-Tashi et al., 2019; Al-Tashi et al., 2020). These algorithms therefore offer distinct search dynamics for the same 561-dimensional binary problem.

A fair comparison requires the classifier and data partition to remain fixed. Support vector machines (SVMs) are appropriate for this purpose because margin maximization and kernel mappings provide strong baselines in high-dimensional spaces (Cervantes et al., 2020). With a radial basis function (RBF) kernel, an SVM can represent nonlinear class boundaries without changing the input features. Holding the kernel and its hyperparameters constant isolates the effect of the selected subset, while avoiding a joint feature-and-model search that would confound the comparison (Gold and Sollich, 2003).

Many HAR studies emphasize a new model or report only the final accuracy. That practice makes it difficult to determine whether a gain comes from a genuinely informative subset, a larger optimization budget, or a favorable classifier setting. Comparative studies in other engineering optimization problems likewise show that algorithm rankings can change when accuracy, convergence, and runtime are considered together (Garip et al., 2021). For HAR, the unresolved question is not merely which optimizer reaches the highest score, but which one provides the most defensible balance between predictive gain, subset size, and search cost.

This study addresses that question through a controlled comparison of GA, PSO, and GWO wrappers coupled with the same RBF-SVM. The analysis uses the original UCI HAR train-test split, a common population size and search horizon, and a shared accuracy-dominant objective with a small penalty for the proportion of selected features. Evaluation extends beyond aggregate accuracy to weighted F1, absolute test errors, class-level confusion, convergence behavior, dimensionality reduction, and elapsed runtime.

The study makes three contributions. First, it quantifies the effect of three different binary search mechanisms on the same SVM-based HAR task. Second, it links overall performance to the two persistent sources of error in this dataset: confusion between walking upstairs and downstairs, and confusion among sitting, standing, and laying. Third, it interprets feature reduction and optimization time together, which prevents the smallest subset from being treated automatically as the best result. These contributions are empirical and comparative rather than methodological.

2. Materials and Methods

This section describes the dataset and controlled experimental design, formulates feature selection as a binary wrapper problem, and then presents the SVM evaluator and the three metaheuristic search procedures.

2.1. Dataset and Experimental Design

The experiments used the Human Activity Recognition Using Smartphones dataset from the UCI Machine Learning Repository. Thirty volunteers aged 19-48 years carried a Samsung Galaxy S II smartphone at the waist while performing walking, walking upstairs, walking downstairs, sitting, standing, and laying. Triaxial acceleration and angular velocity were sampled at 50 Hz. After noise filtering and segmentation into 2.56 s windows with 50% overlap, the dataset developers derived 561 time- and frequency-domain features from each observation (Anguita et al., 2013).

The original subject-independent partition was retained: 7,352 observations formed the training set and 2,947 observations formed the test set. The same data checks and normalization procedure were applied to all methods. Candidate feature masks were scored using validation information drawn only from the training data; the test set was reserved for the final assessment of

the baseline and the three selected subsets. This separation prevented the optimizers from using test labels during feature search.

Feature selection was represented as a 561-dimensional binary optimization problem. For a candidate vector x, x_j=1 indicated that feature j was supplied to the classifier, and x_j=0 indicated that it was omitted. The number of possible masks makes an exhaustive search infeasible, so each metaheuristic maintained a small population of candidate vectors and improved them using its own update mechanism.

The wrapper objective was designed to favor predictive performance while applying a smaller penalty to subset size. The accuracy term was assigned a weight of 0.99, and the selected-feature ratio was assigned a weight of 0.01. This weighting reflects the study objective: dimensionality should be reduced without accepting a substantial loss in recognition performance. The fixed design was chosen to compare the optimizers under the same classifier, data split, population size, search horizon, and objective function rather than to estimate their asymptotic performance.

For every candidate mask, the selected columns were extracted from the training data and evaluated with the same SVM using an RBF kernel. Validation accuracy supplied the predictive component of the objective. Keeping the evaluator fixed ensured that differences among candidate scores arose from feature composition rather than from changing classifier settings. The experiments were conducted in Google Colab with Python 3.12. The principal libraries were NumPy 1.26.0, Pandas 2.2.2, Matplotlib 3.10.0, scikit-learn, PyGAD 3.6.0, and MEALPY 3.0.3. PyGAD was used for GA, MEALPY for PSO, and scikit-learn SVC for classification; GWO was implemented manually. Although an NVIDIA T4 GPU was available in the Colab environment, all reported experiments were executed on the CPU because no GPU-parallel computation was used. Multiprocessing was not enabled. Because the GA implementation treats larger scores as better, its fitness function was written in maximization form as shown in equation 1.

$$Fit_{GA} = 0.99\,Acc - 0.01\,FR \quad (1)$$

The PSO and GWO implementations treat smaller objective values as better. Their common cost function therefore combined validation error with the selected-feature ratio, as given in equation 2.

$$f_{PSO/GWO} = 0.99\,(1 - Acc) + 0.01\,FR \quad (2)$$

In equations 1 and 2, Acc denotes validation accuracy and FR is the number of selected features divided by 561. Fit_GA is the fitness value maximized by GA, whereas f_PSO/GWO is the cost minimized by PSO and GWO. Figures 1, 2, and 3 summarize the sensing pipeline, wrapper loop, and common experimental workflow.

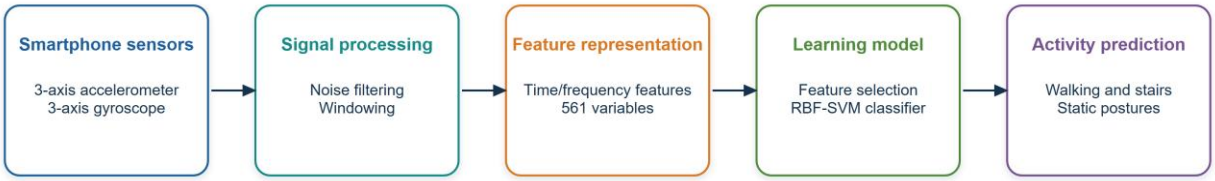

Figure 1. Smartphone sensing and feature-engineering pipeline for the UCI HAR dataset.

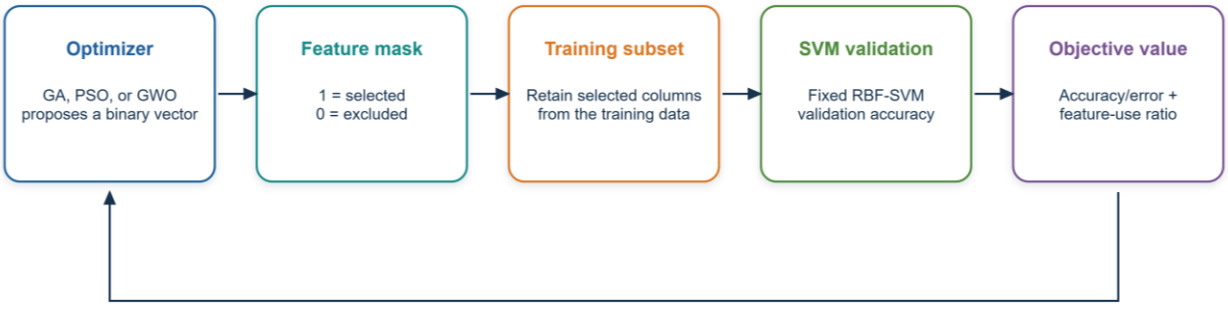

Figure 2. Wrapper-based binary feature-selection loop with a fixed SVM evaluator.

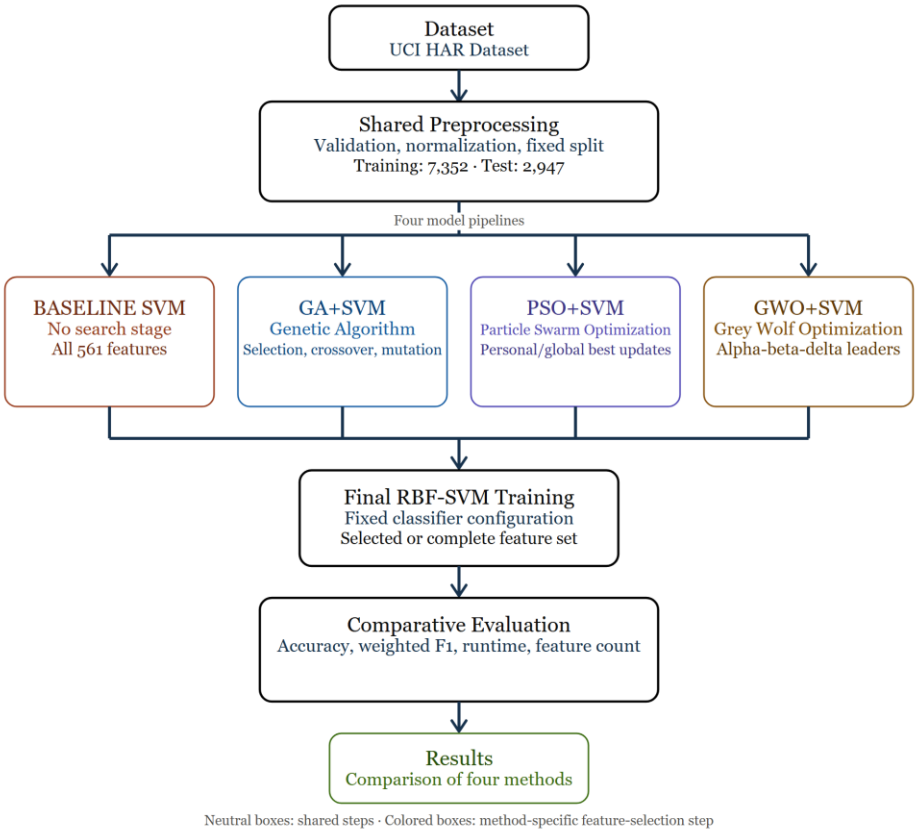

Figure 3. Controlled workflow used to compare the baseline and three feature-selection methods.

2.2. SVM Evaluator and Performance Measures

A support vector machine with an RBF kernel was used as the common evaluator and final classifier. The configuration was fixed at C=1.0 and gamma='scale'. The multiclass problem was handled by the library's standard decomposition, and no optimizer was allowed to tune SVM hyperparameters. The baseline received all 561 variables; GA+SVM, PSO+SVM, and GWO+SVM received only the variables indicated by the final mask.

Held-out performance was reported as accuracy and weighted F1. Accuracy provides the proportion of correct predictions, whereas weighted F1 summarizes class-specific precision and recall while accounting for class support. The analysis also recorded the number and percentage of retained features, the reduction relative to the 561-feature baseline, the total test errors, the convergence history, and the wall-clock runtime. Confusion matrices were inspected to identify whether improvements were distributed across activities or concentrated in particular class pairs.

2.3. Metaheuristic Feature-Selection Methods

All three search methods operated on the same binary representation and were evaluated by the same SVM. Each run used a population of 10 and a horizon of 10 generations or iterations. This deliberately limited budget enabled a controlled comparison of search behavior and computational cost. A population of 10 and a horizon of 10 generations or iterations were retained as a deliberately constrained, matched computational budget. This setting supports a controlled fixed-budget comparison, but it does not characterize the full search behavior or asymptotic performance of the stochastic optimizers.

2.3.1. Genetic Algorithm

Each GA individual was a 561-bit chromosome. Steady-state selection retained four parents, one-point crossover combined parental masks, and random mutation altered 5% of the genes. Two parents were preserved between generations to reduce the risk of losing the best observed subset. The procedure maximized equation 1 for 10 generations. Table 1 lists the implementation parameters.

Table 1. Genetic algorithm parameters

Parameter	Value
Population size	10
Number of generations	10
Number of parents	4
Chromosome length	561
Gene values	0 / 1
Selection	Steady-state selection
Crossover	One-point crossover
Mutation	Random mutation
Mutation rate	5%
Elitist parents retained	2

2.3.2. Particle swarm optimization

PSO was implemented using the OriginalPSO procedure in the MEALPY library, with a binary decision space. Each particle encoded a feature mask and was updated from its personal best position and the best position identified by the swarm. Candidate masks were scored by equation 2, so improvements required either a lower validation error or a smaller subset without a compensating accuracy loss. Table 2 gives the settings.

Table 2. Particle swarm optimization parameters

Parameter	Value
Algorithm	OriginalPSO
Library	MEALPY
Problem type	Binary feature selection
Decision space	BinaryVar
Population size	10
Iterations	10
Objective direction	Minimization
Evaluation model	SVM
SVM kernel	RBF
Evaluation data	Training/validation split
Test metrics	Accuracy, weighted F1, feature ratio, runtime

2.3.3. Grey wolf optimization

GWO was implemented as a binary search procedure. At each iteration, candidate wolves were ranked and the three best masks were designated alpha, beta, and delta. The remaining masks were updated with respect to these leaders, while the control parameter decreased over the run to shift the search from exploration toward exploitation. Binary decisions were recovered after position updates, and equation 2 determined the ranking. Table 3 summarizes the configuration.

Table 3. Grey wolf optimization parameters

Parameter	Value
Algorithm	GWO
Implementation	Custom implementation
Problem type	Binary feature selection
Representation	0 / 1 binary vector
Number of wolves	10
Iterations	10
Accuracy weight (alpha)	0.99
Feature-ratio weight (beta)	0.01
Objective direction	Minimization
Evaluation model	SVM
SVM kernel	RBF
Data partition	Training / validation / test
Test metrics	Accuracy, weighted F1, feature ratio, runtime

3. Results

The baseline and three feature-selected models were evaluated on the same 2,947 test observations. All metaheuristic variants exceeded the baseline accuracy, but the magnitude of improvement, subset size, convergence pattern, and runtime differed substantially.

3.1. Baseline SVM

The 561-feature SVM achieved 0.9308 accuracy and 0.9304 weighted F1, corresponding to 204 errors. Walking was recognized perfectly (537/537). Most dynamic-activity errors involved walking upstairs and downstairs: 68 upstairs observations were predicted as downstairs and 45 downstairs observations were predicted as upstairs. Among postural activities, 33 standing observations were assigned to sitting, while laying was occasionally confused with sitting or standing. The baseline confusion matrix is shown in Figure 4.

3.2. GA+SVM

As shown in Figures 5 and 6, the GA-optimized SVM model achieved strong classification performance, while the convergence curve demonstrates the effectiveness and stability of the genetic algorithm during the optimization process. GA+SVM delivered the best aggregate result.

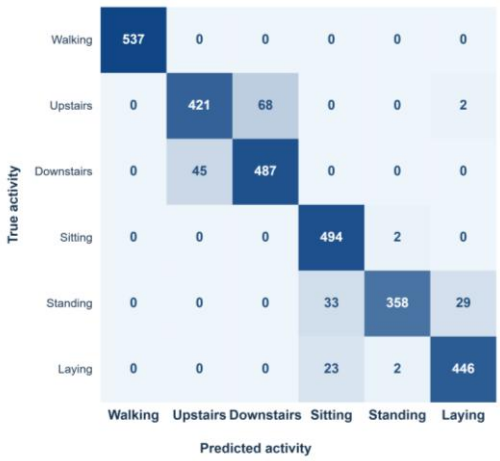

Figure 4. Confusion matrix for the baseline SVM using all 561 features.

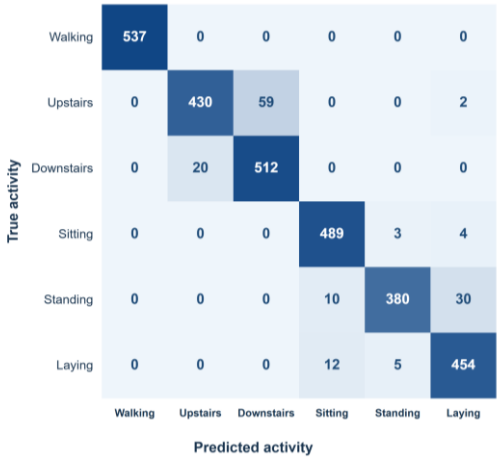

Figure 5. GA+SVM confusion matrix.

It retained 300 features, reducing the original representation by 46.52%, and achieved 0.9508 accuracy with a weighted F1 of 0.9506. The number of test errors fell from 204 to 145, a reduction of 28.92%. The main gain came from more reliable separation of the two stair activities and from fewer confusions among sitting, standing, and laying.

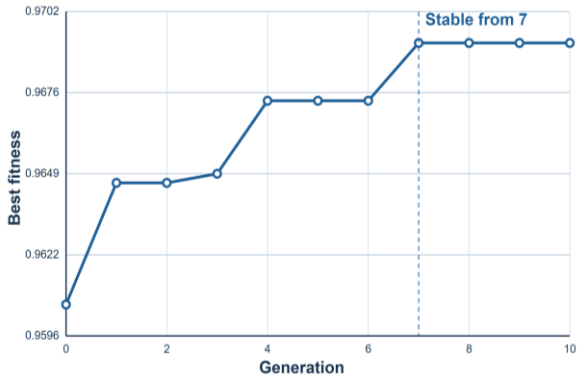

Figure 6. GA convergence curve.

3.3. PSO+SVM

Figures 7 and 8 show the confusion matrix and convergence history results for PSO+SVM. PSO+SVM selected 284 features, the smallest subset in the comparison, and a 49.38% reduction from the baseline. Its accuracy was 0.9355, and its weighted F1 was 0.9354, yielding 190 errors. The method therefore improved on the full-feature SVM, but by a smaller margin than GA+SVM and GWO+SVM.

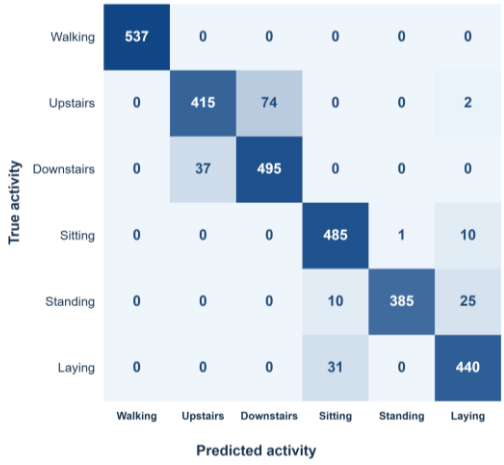

Figure 7. PSO+SVM confusion matrix.

The PSO subset reduced some postural confusions, particularly standing-to-sitting errors, but increased ambiguity between walking upstairs and downstairs compared with GA and GWO. Thirty-one laying samples were also classified as sitting. These errors explain why the smallest subset did not translate into the best predictive result.

The best PSO objective remained at 0.0367 throughout the 10 recorded iterations. Under this population and iteration budget, the swarm did not improve on its initial best region. PSO+SVM also required 489.68 s, the longest runtime in the study.

3.4. GWO+SVM

Figures 9 and 10 show the confusion matrix and convergence history results for GWO+SVM. GWO+SVM retained 341 features, corresponding to a 39.22% reduction, and reached 0.9471 accuracy with a weighted F1 of 0.9470. Its 156 test errors were 23.53% fewer than the baseline. Although it selected more features than GA and PSO, its predictive performance was second only to GA. GWO produced the fewest combined errors between walking upstairs and downstairs: 55 upstairs samples were assigned to downstairs, and 22 downstairs samples were assigned to upstairs. Its postural errors were less favorable than GA's, particularly for laying-to-sitting decisions. This class-level pattern accounts for the small aggregate gap between the two methods.

The GWO objective improved twice: from 0.02815 initially to 0.02710, then to 0.02558, after which it remained stable from iteration 5 onward. Its 295.12 s runtime placed it between GA and PSO.

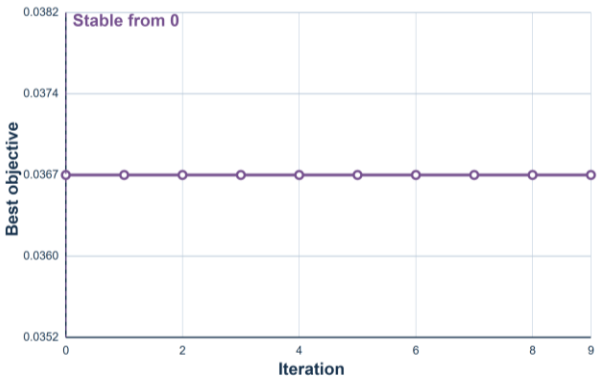

Figure 8. PSO convergence curve.

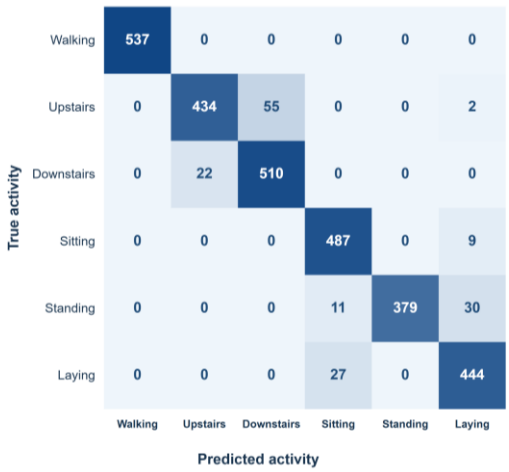

Figure 9. GWO+SVM confusion matrix.

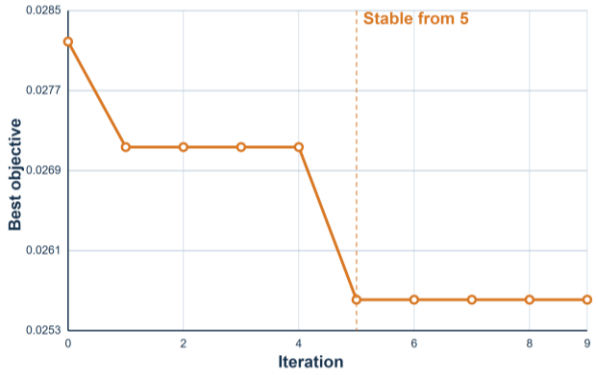

Figure 10. GWO convergence curve.

3.5. Comparative Analysis

Table 4 consolidates predictive performance, feature reduction, convergence, and runtime. Figure 11 presents a trade-off view in which accuracy and dimensionality reduction are considered alongside optimization time. The comparison identifies three outcomes: GA produced the strongest classifier, PSO produced the smallest subset, and GWO achieved the second-highest accuracy.

Relative to the baseline, accuracy increased by 0.0200 with GA+SVM, 0.0163 with GWO+SVM, and 0.0047 with PSO+SVM. PSO selected 16 fewer features than GA, yet its accuracy was 0.0153 lower, and its search took approximately 2.25 times as long. GWO was 0.0037 less accurate than GA, retained 41 additional features, and required about 1.35 times the runtime. Under the tested conditions, GA therefore occupied the most favorable region of the accuracy-reduction-runtime trade-off. The 0.9508 accuracy is competitive within the tested UCI HAR and fixed RBF-SVM configuration, particularly relative to the all-feature baseline.

4. Discussion

The results show that feature selection improved the SVM not simply by shrinking the input but by removing variables that interfered with difficult class boundaries. GA+SVM reduced errors in both major confusion groups, whereas the baseline committed more cross-assignments between the stair activities and among the three postures. This pattern is consistent with the rationale for wrapper selection: variables are retained based on their joint contribution to a specified classifier rather than on an isolated univariate score. GA's recombination and mutation appear to have explored useful combinations within the limited budget, although this interpretation is based on the observed run and is not a statistical comparison.

PSO and GWO illustrate why subset size, convergence, and accuracy should be reported together. PSO removed the largest proportion of variables, but its flat convergence curve and long runtime indicate that the additional reduction was not obtained efficiently. GWO improved early and produced a competitive model, especially for upstairs/downstairs discrimination, yet its larger subset and postural errors kept it below GA overall. Binary swarm methods are sensitive to transfer rules, parameter settings, and initial positions (Chuang et al., 2008; Emary et al., 2016); a small common budget exposes these differences but does not fully characterize each algorithm's best attainable performance.

From a deployment perspective, the selected subsets may reduce the cost of feature extraction and subsequent SVM inference, but the search itself is an offline expense. The baseline trained and evaluated in 2.76 s, whereas optimization required 217.99-489.68 s. That cost can be acceptable when a subset is learned once and reused on many devices, but it matters when models must be updated frequently. The study has three main limitations. First, each stochastic method was evaluated from a single recorded run, so variance and statistical significance were not estimated. Second, the population and iteration budgets were intentionally small. Third, SVM hyperparameters were fixed rather than jointly optimized. Accordingly, the observed ranking applies to this dataset split and configuration; it should not be generalized as an inherent superiority of GA over all PSO or GWO variants.

Table 4. Comparative performance of SVM, GA+SVM, PSO+SVM, and GWO+SVM

Method	Accuracy	Weighted F1	Features	Reduction (%)	Convergence	Runtime (s)
SVM	0.9308	0.9304	561	0.00	Not applicable	2.76
GA+SVM	0.9508	0.9506	300	46.52	Stable after generation 7	217.99
PSO+SVM	0.9355	0.9354	284	49.38	No improvement after initialization	489.68
GWO+SVM	0.9471	0.9470	341	39.22	Stable after iteration 5	295.12

Figure 11. Accuracy, feature reduction, and optimization-time trade-off across the four models.

5. Conclusion

This study compared GA, PSO, and GWO as wrapper-based feature selectors for a fixed RBF-SVM on the UCI HAR benchmark. Evaluating the methods under the same partition, classifier, population size, search horizon, and objective made it possible to attribute the differences primarily to their search behavior.

GA+SVM achieved the best observed result with 0.9508 accuracy, 0.9506 weighted F1, and 300 features. It reduced dimensionality by 46.52% and test errors from 204 to 145. GWO+SVM achieved 0.9471 accuracy with 341 features, while PSO+SVM selected the smallest subset but delivered only a modest gain over the baseline and incurred the longest search time.

These findings indicate that the most compact representation is not necessarily the most useful one. For this experiment, GA provided the most balanced combination of recognition performance, subset size, and optimization cost. The remaining errors also identify a practical target for subsequent work: signal descriptors that better separate ascent from descent and distinguish similar static postures.

Future studies should repeat each stochastic search with multiple seeds, report confidence intervals and statistical tests, examine larger and matched computational budgets, and evaluate whether jointly tuning SVM hyperparameters changes the ranking. Validation on additional body locations, devices, and HAR datasets would establish whether the selected feature structures transfer beyond the UCI benchmark. Future work should assess inference latency, memory use, energy consumption, and feature-extraction time on representative mobile devices, while also verifying these findings through multi-seed sensitivity analyses with matched objective-function evaluation counts.

Author Contributions

The percentages of the authors' contributions are presented below. The authors reviewed and approved the final version of the manuscript.

	D.S.Ç.	Ö.K.	D.Ö.Ş.	G.K.	S.D.
C	20	20	20	20	20
D	5	5	25	40	25
S	40	40	10	5	5
DCP	40	40	10	5	5
DAI	25	25	20	15	15
L	40	30	10	10	10
W	25	25	20	10	20
CR	5	5	40	25	25
SR	20	20	20	20	20
PM	20	20	20	20	20

C= concept, D= design, S= supervision, DCP= data collection and/or processing, DAI= data analysis and/or interpretation, L= literature search, W= writing, CR= critical review, SR= submission and revision, PM= project management.

Conflict of Interest

The authors declared that there is no conflict of interest.

Ethical Consideration

Ethics committee approval was not required for this study because there was no study on animals or humans.

Data Availability Statement

The Human Activity Recognition Using Smartphones dataset is publicly available from the UCI Machine Learning Repository at https://archive.ics.uci.edu/dataset/240/human+activity+recognition+using+smartphones.

References

Agrawal, P., Abutarboush, H. F., Ganesh, T., & Mohamed, A. W. (2021). Metaheuristic algorithms on feature selection: A survey of one decade of research (2009–2019). *IEEE Access*, *9*, 26766–26791. https://doi.org/10.1109/ACCESS.2021.3056407

Akinola, O. O., Ezugwu, A. E., Agushaka, J. O., Zitar, R. A., & Abualigah, L. (2022). Multiclass feature selection with metaheuristic optimization algorithms: A review. *Neural Computing and Applications*, *34*(22), 19751–19790. https://doi.org/10.1007/s00521-022-07705-4

Al-Tashi, Q., Kadir, S. J. A., Rais, H. M., Mirjalili, S., & Alhussian, H. (2019). Binary optimization using hybrid grey wolf optimization for feature selection. *IEEE Access*, *7*, 39496–39508. https://doi.org/10.1109/ACCESS.2019.2906757

Al-Tashi, Q., Abdulkadir, S. J., Rais, H. M., Mirjalili, S., Alhussian, H., Ragab, M. G., & Alqushaibi, A. (2020). Binary multi-objective grey wolf optimizer for feature selection in classification. *IEEE Access*, *8*, 106247–106263. https://doi.org/10.1109/ACCESS.2020.3000040

Anguita, D., Ghio, A., Oneto, L., Parra, X., & Reyes-Ortiz, J. L. (2013, April 24–26). *A public domain dataset for human activity recognition using smartphones* [Paper presentation]. 21st European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN 2013), Bruges, Belgium. https://www.esann.org/sites/default/files/proceedings/legacy/es2013-84.pdf

Babatunde, O. H., Armstrong, L., Leng, J., & Diepeveen, D. (2014). A genetic algorithm-based feature selection. *International Journal of Electronics Communication and Computer Engineering*, *5*(4), 899–905. https://ro.ecu.edu.au/ecuworkspost2013/653/

Budak, H. (2018). Özellik seçim yöntemleri ve yeni bir yaklaşım. *Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, *22*(1), 21–31. https://izlik.org/JA45SW33BN

Cervantes, J., Garcia-Lamont, F., Rodríguez-Mazahua, L., & Lopez, A. (2020). A comprehensive survey on support vector machine classification: Applications, challenges and trends. *Neurocomputing*, *408*, 189–215. https://doi.org/10.1016/j.neucom.2019.10.118

Chuang, L. Y., Chang, H. W., Tu, C. J., & Yang, C. H. (2008). Improved binary PSO for feature selection using gene expression data. *Computational Biology and Chemistry*, *32*(1), 29–38. https://doi.org/10.1016/j.compbiolchem.2007.09.005

Clarkson, B. P. (2002). *Life patterns: Structure from wearable sensors* (Doctoral dissertation, Massachusetts Institute of Technology). DSpace@MIT. http://hdl.handle.net/1721.1/8030

Dentamaro, V., Gattulli, V., Impedovo, D., & Manca, F. (2024). Human activity recognition with smartphone-integrated sensors: A survey. *Expert Systems with Applications*, *246*, Article 123143. https://doi.org/10.1016/j.eswa.2024.123143

Emary, E., Zawbaa, H. M., & Hassanien, A. E. (2016). Binary grey wolf optimization approaches for feature selection. *Neurocomputing*, *172*, 371–381. https://doi.org/10.1016/j.neucom.2015.06.083

Garip, Z., Çimen, M. E., & Boz, A. F. (2021). Meta-sezgisel algoritmalar kullanarak güneş pili modellerinin parametre çıkarımında karşılaştırmalı performans analizi. *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, *36*(2), 1133–1144. https://doi.org/10.17341/gazimmfd.586269

Gazioğlu, E. (2024). Dinamik özellik seçimi problemlerinde meta-sezgisel yaklaşımlar. In *Bilgisayar bilimleri ve mühendisliğinde öncü ve yenilikçi çalışmalar* (pp. 46–61). All Sciences Academy Yayınları. https://www.allsciencesacademy.com/_files/ugd/13252f_13a3215bdc3241a19c52faa5bd2766ea.pdf

Gold, C., & Sollich, P. (2003). Model selection for support vector machine classification. *Neurocomputing*, *55*(1–2), 221–249. https://doi.org/10.1016/S0925-2312(03)00375-8

Hussein, F., Kharma, N., & Ward, R. (2001). *Genetic algorithms for feature selection and weighting: A review and study*. Proceedings of the Sixth International Conference on Document Analysis and Recognition (ICDAR 2001), Seattle, WA, USA. https://doi.org/10.1109/ICDAR.2001.953980

Kwapisz, J. R., Weiss, G. M., & Moore, S. A. (2011). Activity recognition using cell phone accelerometers. *ACM SIGKDD Explorations Newsletter*, *12*(2), 74–82. https://doi.org/10.1145/1964897.1964918

Oh, I. S., Lee, J. S., & Moon, B. R. (2004). Hybrid genetic algorithms for feature selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *26*(11), 1424–1437. https://doi.org/10.1109/TPAMI.2004.105

Tang, J., Alelyani, S., & Liu, H. (2014). Feature selection for classification: A review. In C. C. Aggarwal (Ed.), *Data classification: Algorithms and applications* (pp. 37–64). CRC Press.

Tokgöz, N., & Kocaoğlu, S. (2026). An IMU-based dataset of falls, activities of daily living, and prayer movements (AybuFall). *BMC Research Notes*, *19*, Article 78. https://doi.org/10.1186/s13104-026-07679-9

Tonguç, G. (2024). Optimizasyon algoritmalarına genel bakış: Klasik ve modern yöntemler. *Kuantum Teknolojileri ve Enformatik Araştırmaları Dergisi*, *2*(3), 105–113. https://doi.org/10.70447/ktve.2561

Vurgun, Y., & Kıran, M. S. (2024). İnsan aktivite tanıması için yeni bir veri kümesi ve derin öğrenme modelleri ile sınıflandırılması. *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, *40*(1), 653–672. https://doi.org/10.17341/gazimmfd.1325926

Xue, B., Zhang, M., & Browne, W. N. (2013). Particle swarm optimization for feature selection in classification: A multi-objective approach. *IEEE Transactions on Cybernetics*, *43*(6), 1656–1671. https://doi.org/10.1109/TSMCB.2012.2227469

Xue, B., Zhang, M., Browne, W. N., & Yao, X. (2016). A survey on evolutionary computation approaches to feature selection. *IEEE Transactions on Evolutionary Computation*, *20*(4), 606–626. https://doi.org/10.1109/TEVC.2015.2504420