



Home Automation-Based Health Assessment Along Gesture Recognition via Inertial Sensors

Hammad Rustam¹, Muhammad Muneeb¹, Suliman A. Alsuhibany², Yazeed Yasin Ghadi³,
Tamara Al Shloul⁴, Ahmad Jalal¹ and Jeongmin Park^{5,*}

¹Department of Computer Science, Air University, Islamabad, 44000, Pakistan

²Department of Computer Science, College of Computer, Qassim University, Buraydah, 51452, Saudi Arabia

³Department of Computer Science and Software Engineering, Al Ain University, Al Ain, 15551, UAE

⁴Department of Humanities and Social Science, Al Ain University, Al Ain, 15551, UAE

⁵Department of Computer Engineering, Korea Polytechnic University, 237 Sangidaehak-ro Siheung-si, Gyeonggi-do, 15073, Korea

*Corresponding Author: Jeongmin Park. Email: jmpark@kpu.ac.kr

Received: 16 February 2022; Accepted: 19 April 2022

Abstract: Hand gesture recognition (HGR) is used in a numerous applications, including medical health-care, industrial purpose and sports detection. We have developed a real-time hand gesture recognition system using inertial sensors for the smart home application. Developing such a model facilitates the medical health field (elders or disabled ones). Home automation has also been proven to be a tremendous benefit for the elderly and disabled. Residents are admitted to smart homes for comfort, luxury, improved quality of life, and protection against intrusion and burglars. This paper proposes a novel system that uses principal component analysis, linear discrimination analysis feature extraction, and random forest as a classifier to improve HGR accuracy. We have achieved an accuracy of 94% over the publicly benchmarked HGR dataset. The proposed system can be used to detect hand gestures in the healthcare industry as well as in the industrial and educational sectors.

Keywords: Genetic algorithm; human locomotion activity recognition; human-computer interaction; human gestures recognition principal; hand gestures recognition; inertial sensors; principal component analysis; linear discriminant analysis; stochastic neighbor embedding

1 Introduction

Gesture recognition technology has a variety of applications, including health-care. This is something that a lot of researchers and businesses are interested in. According to MarketsandMarkets' analysis, the health-care sector has the potential to become an important industry for gesture-based recognition technologies over the next five years. In health-care, gesture recognition technology is categorized into two techniques: gesture recognition using wearable sensors and computer vision. A computer-vision-based hand gesture recognition method proposed by Wachs et al. [1] despite the



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

fact that the system has been examined in real-life situations, it still has certain flaws. It is expensive, demands appropriate color calibration, and is impacted by lighting. Wearable handglove (sensor embedded) gesture recognition technology, in comparison to computer vision-based recognition, is less costly, consumes low power, requires only light-weight processing, does not require prior color calibration, does not violate users' privacy, and is not affected by lighting environment.

The Internet-of-Things (IoT) is a field in which certain appliances are integrated with electronic devices, software, and sensors and exchange data with other connected devices to serve the users. Gonzalo et al. [2] presented important research on the IoT to improve the user experience when working with smart home devices. This focuses on user-device interaction. Many IoT-related solutions have been developed, according to Pavlovic et al. [3] in order to provide communication capabilities to common things, as a feasible technique for communicating with home automation systems. The concept of a hand gesture is the basic idea that is associated with action, reaction, response, or a necessity that the user wants to fulfill. As a result, hand gestures play an essential role in operating home automation systems. Vision-based and sensor-based gesture recognition are two types of gesture recognition. Hand gesture recognition is supported by computer vision, which analyses images gained from video [4], as well as sensor-based gesture recognition, which analyses signals received from smart phones with sensors [5]. Inertial sensors such as accelerometers and gyroscopes have proven useful for HLAR [6]. The increased usage of unmanned aerial vehicles (UAVs) in the transportation systems facilitates the development of object identification algorithms for collecting real-time traffic data using UAVs.

The basic sensor for human locomotion activity recognition (HLAR) is a three-axis accelerometer. This system is capable of capturing acceleration along the three axis x, y, and z [7]. When it comes to smartphones, position is a big deal [8] because each individual has their own style of keeping their smartphone, either in a dress pocket, in a purse, or carried in hand. The former, on the other hand, can generate privacy problems, lighting circumstances, and space limitations. But the latter, on the other hand, alleviates privacy concerns as well as environmental difficulties like light and space. Furthermore, the latest technological devices have influenced the latter approach. Mobile phone sensors such as accelerometers, gyroscopes, and magnetometers are shrinking in size and becoming wireless, allowing them to be readily integrated into smart devices (phones and watches) and household appliances (fans, bulbs, air-conditioners). These sensors can detect complicated motions and human behavior in addition to capturing sensor signals such as acceleration, rotation, and direction. In particular, gesture recognition, which can be applied to home automation to control home appliances, is an important framework for smart-homes.

There are three main contributions to this work:

- The proposed system of automatic hand gestures recognition can be used in the context of home automation.
- A dataset of nine gestures that are easy to memorize by the user and are suitable for controlling devices in smart homes.
- Users can use this system to easily control multiple devices such as fans and bulbs.

The article has the following sections: Section 2 explains related work that has been done in the past by different researchers, which is divided into two further parts. Section 3 includes our proposed methodology for our system, which involves preprocessing along with dimension reduction and feature extraction. Genetic algorithm is used in this section for feature selection. Section 4 discusses the experiments and their conclusion.

2 Related Work

The related work is divided into two subsections medical assessment based on inertial sensors and medical assessment based on video sensors systems proposed in recent years.

2.1 Medical Assessment via Inertial Sensors

Human-computer interaction (HCI) has recently attracted considerable attention. As a result, within the proactive computing discipline, where knowledge is actively presented, technologies for recognizing the intention of the user have been actively investigated. The utilization of gesture recognition has been the focus of such research [9]. Batool et al. [10] published the results of a worldwide survey of gesture-based user interfaces to support a comfortable and relaxed interface among users and different devices.

Several studies and applications using gesture recognition have previously been carried out, utilizing a variety of technologies. Amin et al. [11] describe a continuous hand gesture (CHG) detection technique in such smart gadgets using 3-axis gyroscope and accelerometer sensors. A smart device gets sensor data from the inertial sensors during the data acquisition phase. After that, the raw sensory data is delivered into the preprocessing processes, which compress and code it. During the construction process, they established a gesture database to recognize hand motions. The start and finish locations of each gesture are automatically identified by the gesture recognition procedure. The gesture reorganization stage is finally processed by the coded sensory data.

2.2 Medical Assessment via Vision Sensors

Static and dynamic gestures are classified as two different types of gestures. The system recognizes a gesture after a spurt of time, which is known as a static gesture, while a dynamic gesture changes in a time frame. The camera sensor is used for computer vision applications so that when an individual performs a gesture, it perceives an image and sends it to the active system so that it may perform recognition [12]. Artificial intelligence is becoming increasingly significant in the field of education. In 2001, people were wondering whether, it was possible to achieve such an environment where machines were teaching humans, but with the advancement of Artificial Intelligence (AI), it is possible now. With advancements in HGS, there is space to develop such a tool that will be sustainable and reliable for all of the current world-increasing schemes of e-learning models [13,14]. There are several ways to detect hand gestures, such as using Electromyography (EMG) [15], cameras [16], or wearable data gloves [17]. A lot of developments in recent technologies for Human Activity Recognition (HAR) use computer vision by using depth maps, which is the application of Artificial Intelligence without using any motion sensors or any markers [18] with time and advancements, HAR can be improved a lot better with depth video sensors [19]. Any activity which is done through in-depth video-based human-activity-recognition (HAR) has a lot of contributions to smart homes and healthcare services [20]. One of the most promising applications in the field of computer vision is vehicle re-identification (Re-ID). The proposed TBE-Net combines complementary traits, global appearance, and local area elements into a unified framework for subtle feature learning, resulting in more integrated and diverse vehicle attributes that can be utilized to distinguish the vehicle from others.

3 Material and Methods

Our system design comprises of the following components: Inertial sensors handle data gathering for gestures. The following has two major tasks: (i) collect data from inertial sensors and transmit it to be preprocessed; and (ii) send the data to be preprocessed. We performed PCA to reduce the

dimensionality of the data before proceeding now to feature extraction. Feature extraction takes a collection of observable data and transforms it into a useful source (feature) that is both informative and non-redundant, making learning and extrapolation easier. Feature extraction's purpose is to enable us to use these features to distinguish between different sorts of objects. Feature extraction is the process of lowering the resources needed to explain a huge amount of data. As a feature extraction method, we employ Linear Discriminant Analysis (LDA). To avoid overfitting, the goal is to reduce the computational cost of a dataset by projecting it to a lower dimensional space with excellent class separation. The t-SNE feature extraction method, it is capable of capturing the large bulk of the spatial relationship of high-dimensional data, is the second feature extraction method we applied. Euclidean distances between data points in high dimensions are transformed into conditional probabilities that show similarities in Stochastic Neighbor Embedding (t-SNE). To get excellent accuracy, features must be processed by a genetic algorithm (GA) after feature reduction. Genetic algorithm is a general-purpose searching approach that has been used to solve optimization problems successfully. As depicted in Fig. 1, GA begins with the random initialization of the population known as chromosomes.

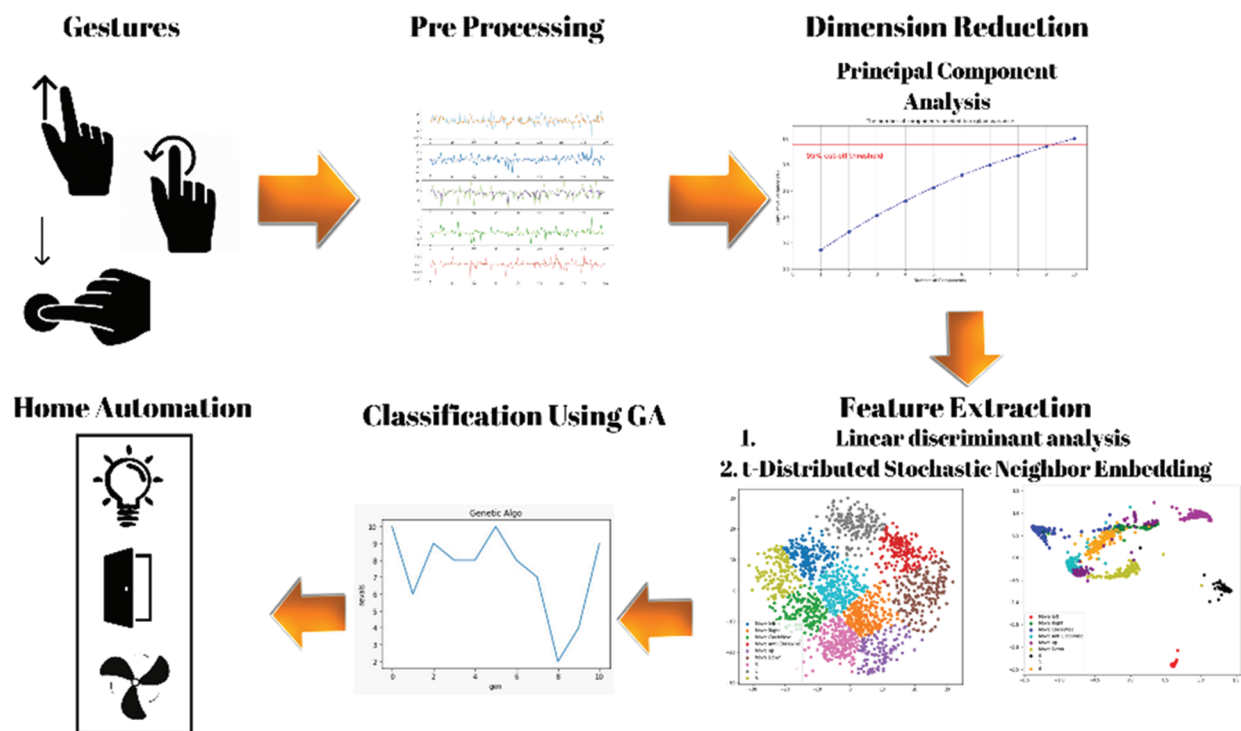


Figure 1: Shows complete flow of the proposed system architecture

3.1 Preprocessing and Dimension Reduction

In data preprocessing, it is essential to identify and deal with the missing values correctly; otherwise, you may draw faulty conclusions and inferences from the data. Therefore, the data preprocessing technique is applied to a dataset that transforms raw data into an understandable and readable format. Fig. 2 shows the filtered and unfiltered data present in the HGR dataset.

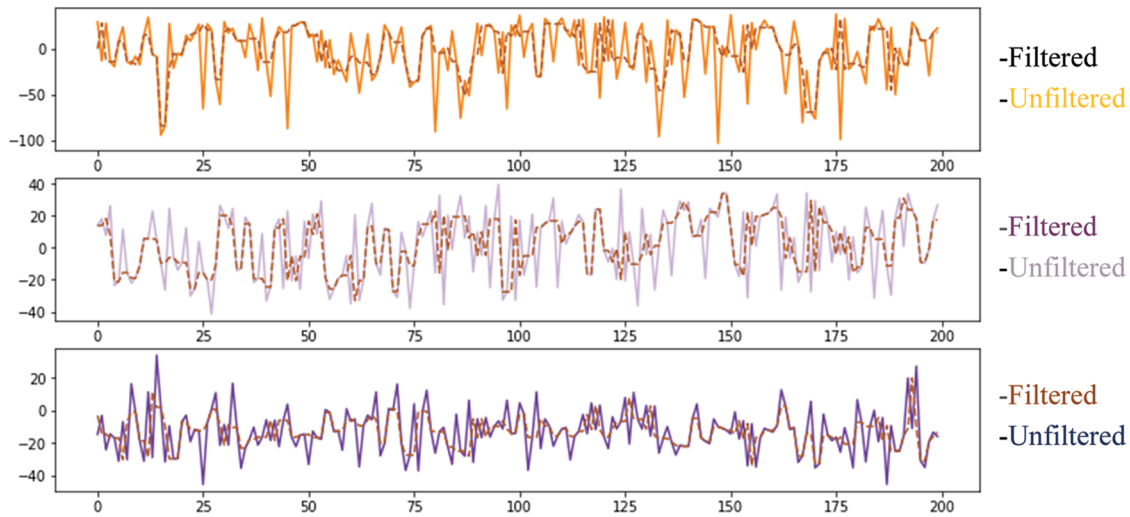


Figure 2: Shows the filtered data of the HGR features

By definition, dimension reduction is the process of converting a dataset of high dimension into a dataset of low dimension that retains all of the original data's features. It is the procedure for creating a set of uncorrelated primary variables to limit the number of random variables to be considered [21]. PCA is used in our system to reduce the data dimension. The data is collected and analyzed by the PCA method, which is widely used to minimize the dimensions of a dataset while maintaining the characteristics that contribute the most to variance. To retrieve the main components (eigenvectors) and their weights, we utilize PCA to eigendecompose the covariance matrix (eigenvalues). Then, by deleting the components that correlate to the smallest eigenvalues, we'll have the best low-dimensional data with the least amount of information lost. The following are some of the most important properties of PCA [22]:

- Retain the original variable information if you want to be representative.
- Separation: There is no overlap between the basic components.
- Compactness: A small number of components replace the original large number of variables.

$T = XW$ changes the space of p variables in a data vector x to a new space of uncorrelated p variables over the dataset. Not all of the major components, however, must be kept. Simply the first L principal components, which are derived by only using the first L eigenvectors, are kept in the truncated transformation as shown in Fig. 3.

$$T_L = XW_L \quad (1)$$

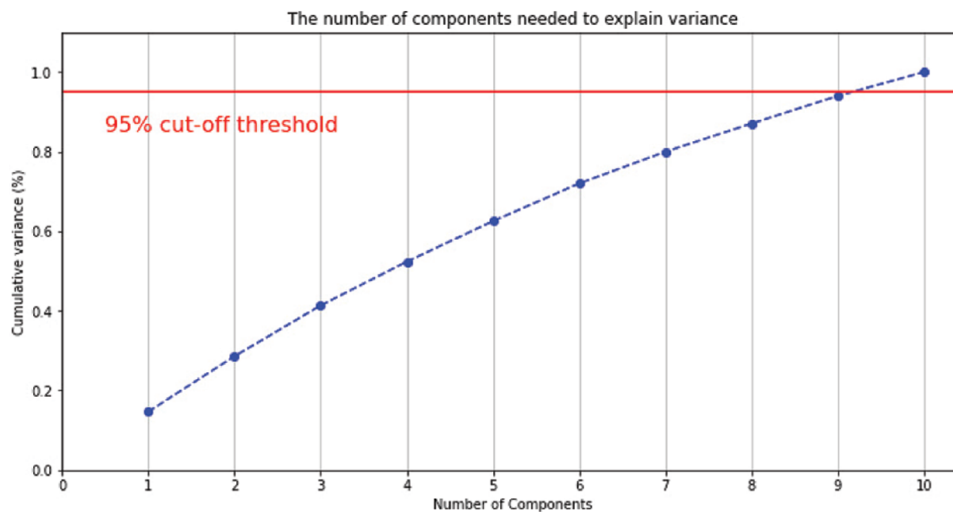


Figure 3: PCA threshold value 95% cut off threshold

3.2 Feature Extraction

3.2.1 LDA

The proposed technique begins with a set of observed data and provides important and non-redundant derived values (features) that enhance learning and extrapolation. Feature extraction enables you to use these properties to distinguish between different types of objects. Feature extraction is a method for reducing the time and resources needed to explain a large set of data because a huge number of variables necessitates a lot of processing and memory resources, a classification method may overfit the training set and undergeneralize to new data. Extraction of features is a broad phrase that refers to methods of combining variables in order to avoid these problems while accurately recording data. LDA is commonly used for the pre-processing step of pattern recognition and machine learning approaches as a feature extraction tool. Data is projected onto a reduced dimension with adequate class separability to avoid overfitting and reduce computing costs.

- Calculating the separability, between-class variance is the term for this:

$$S_b = \sum_{i=1}^g N_i (\bar{X}_i - \bar{X})(\bar{X}_i - \bar{X})^T \quad (2)$$

- To determine the distance between each class's mean and sample:

$$S_w = \sum_{i=1}^g (N_i - 1) S_i = \sum_{i=1}^g \sum_{j=1}^{N_i} (X_{ij} - \bar{X}_i)(X_{ij} - \bar{X}_i)^T \quad (3)$$

- Construct a lower-dimensional environment which enhances between-class variance while reducing within-class variance. Fisher's criterion is a space projection with a reduced dimension as shown in the Fig. 4.

$$P_{lda} = \arg_p \max \frac{|P^T S_b P|}{|P^T S_w P|} \quad (4)$$

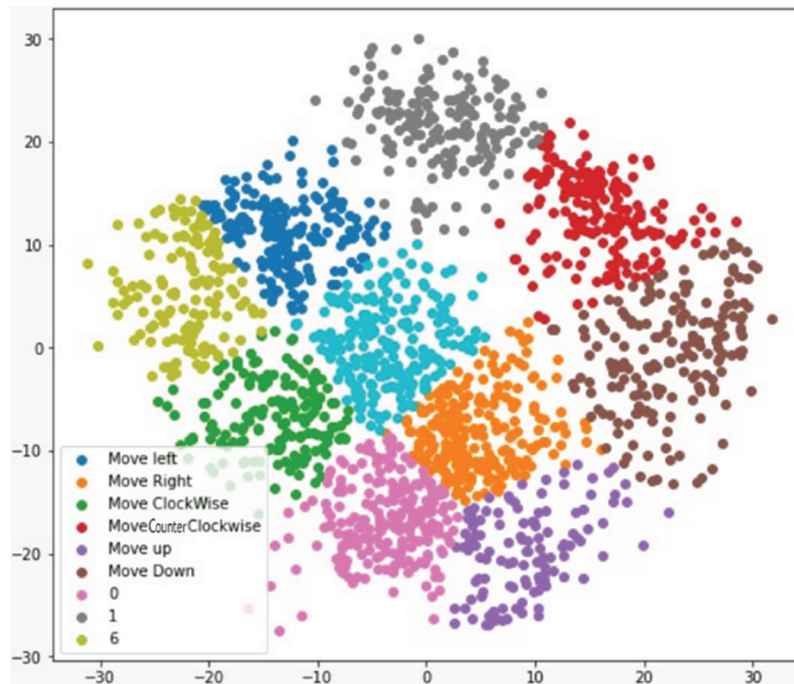


Figure 4: Discriminating 9 classes using linear discrimination analysis

3.2.2 *t-Distributed Stochastic Neighbor Embedding (t-SNE)*

Effective technique in capturing the high-dimensional data is t-SNE. High-dimensional Euclidean distances between data examples are transformed into probable outcomes in t-Stochastic Neighbor Embedding, which reveals commonalities (t-SNE) as shown in the Fig. 5. The conditional probability demonstrated by $P_{j|i}$, and x_i will choose x_j as its neighbor were selected under such a Gaussian centered at x_i is the similarity of the data instances x_j to datapoint x_i . $P_{j|i}$ will be almost infinitesimal (for appropriate values of the Gaussian variance, σ_i) for data points that are widely spaced.

Mathematically, $p_{j|i}$ is given by

$$P_{j|i} = \frac{\exp(-\|x_i - x_j\|^2 / 2\sigma_i^2)}{\sum_{k \neq i} \exp(-\|x_i - x_k\|^2 / 2\sigma_i^2)} \tag{5}$$

$$q_{ij} = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_{k \neq i} (1 + \|y_i - y_k\|^2)} \tag{6}$$

$$\frac{\delta C}{\delta y_i} = 4 \sum_j (p_{ij} - q_{ij}) (y_i - y_j) (1 + \|y_i - y_j\|^2)^{-1} \tag{7}$$

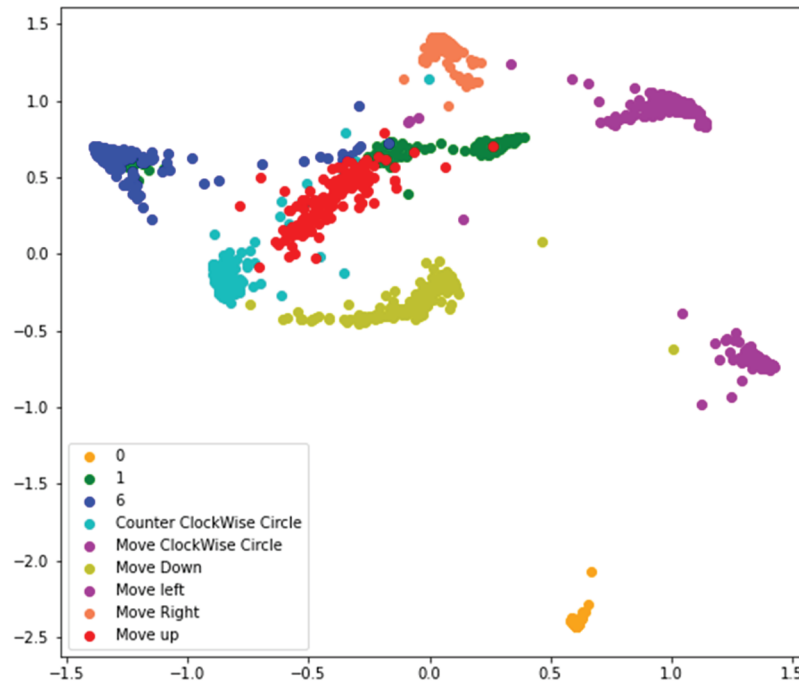


Figure 5: Discriminating 9 classes using t-distributed stochastic neighbor embedding

Algorithm 1: t-SNE Algorithm

Input: $X = (x_1, x_2, x_3)$

Output: Low-dimensional data representation $Y(T) = \{y_1, y_2, \dots, y_n\}$

1: **begin**

2: Compute pairwise affinities p_{ij} with perplexity $Perp$

3: Set $P_{ij} = \frac{P_{ji} + P_{ij}}{2^n}$

4: Sample initial solution $Y^{(0)} = \{y_1, y_2, \dots, y_n\}$ from $N(0, 10^{-4}I)$

5: **For** $t = 1$ to T **do**

6: Compute low-dimensional affinities q_{ij}

7: Compute gradient $\frac{\delta C}{\delta Y}$

8: Set $Y^{(t)} = Y^{(t-1)} + \eta \frac{\delta C}{\delta Y} + \alpha(t) (Y^{(t-1)} - Y^{(t-2)})$

9: **end for**

10: **end**

3.3 Feature Selection: Genetic Algorithm

To achieve better precision, the attributes must be processed by a genetic algorithm after dimension reduction. GA is a standard search strategy that's been used to effectively solve optimization problems. GA begins with the random assignment of chromosomes to a population. Generation is

the continual iteration of chromosomes in a fitness function. The chromosomes are then transmitted through crossover and mutation, resulting in a generation, which is referred to as offspring. Fresh generation chromosomes are picked and rejected depending on fitness function values in the final phase to keep the population size constant. After several iterations, the evolutionary algorithm finds the best optimal solution. For feature classification, we used the GA code from the stages below. (i) String coding, (ii) Initial Population, (iii) Evaluation, and (iv) Genetic Operations.

Algorithm 2: Genetic Algorithm

Input: Feature vectors (v_1, v_2, v_3)

Output: List of features (f_1, f_2, f_3)

```

1: For vector in population labeled do
2:   Features [ ]
3:   While fitness not achieved or fitness not changing do
4:     for features in vectors do
5:       Feature (feature)
6:     end for
7:     Reselect()
8:     child1, child 2 CrossOver (vector)
9:     mutated Mutate (vector)
10:    fitness GetFitness (vector)
11:  end while return Features
12: end for
  
```

A genetic algorithm is given a data partition and uses a cross-validation classification ranking approach to select features. The optimally selected set of functions is shown in the final output at the bottom right. The discriminant analysis methods for all current feature sets are shown in the lower left section of the dashed red box. To establish a fitness-based selection, this was evaluated numerous times for varied training and test data (validation test repeats). In the population repeat loop, this was then used to create a new collection of features. The optimally picked features are displayed in the final output (bottom right) as shown in Fig. 6.

Crossover and mutations are essential for the suggested GA to be stochastic. The crossover of the featured genes is mathematically represented in (5) and (6), where F and \bar{F} are associated with two parent chromosomes, and F represents the crossed child $\times (F, \bar{F})$, as a result of the operation done in (5).

$$\chi(F, \bar{F}) = X_{f1} \dots X_{fn}, Y_{f1} \dots Y_{fn}, Z_{f1} \dots Z_{fn} \overset{\times}{\leftrightarrow} \bar{X}_{f1} \dots \bar{X}_{fn}, \bar{Y}_{f1} \dots \bar{Y}_{fn}, \bar{Z}_{f1} \dots \bar{Z}_{fn} \quad (8)$$

$$\acute{F} = \chi(F, \bar{F}) = \acute{X}_{f1} \dots \acute{X}_{fn}, \acute{Y}_{f1} \dots \acute{Y}_{fn}, \acute{Z}_{f1} \dots \acute{Z}_{fn} \quad (9)$$

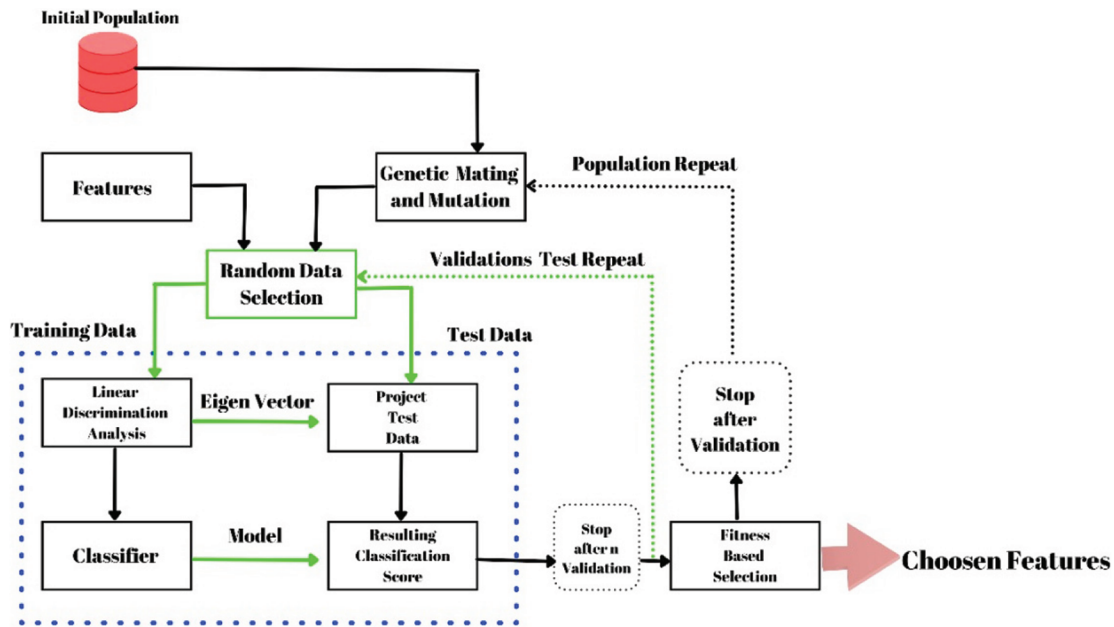


Figure 6: Genetic algorithm feature selection approach is depicted in a flow chart

3.4 Classification for HGR

3.4.1 Decision Tree

Decision tree algorithm begins with the root node of the tree and proceeds to predict the class of a given dataset. It analyzes the value of the root node of the tree with the records attribute, then moves to the branch corresponding to that value and jumps to the next node. It then compared the node’s value to the values of the other sub-nodes until it reached the leaf node of the tree. Information gain refers to a decrease or shift in entropy that occurs after segmenting a dataset depending on an attribute. It assesses how much information a feature provides about a class and tries to obtain more information whenever possible. We split the node and created the decision tree based on the relevance of information gain. It can be calculated using the formula below:

$$IG(Y, X) = E(Y) - E(Y|X) \tag{10}$$

To measure impurity or instability, entropy of a certain property is required. It measures the randomness of information as shown in Fig. 7. The Entropy can be calculated as follows: The Gini coefficient can be calculated using the following formula:

$$E(S) = \sum_{i=1}^c -P_i \log_2 P_i \tag{11}$$

$$\text{Gini Index} = 1 - \sum_j P_j^2 \tag{12}$$

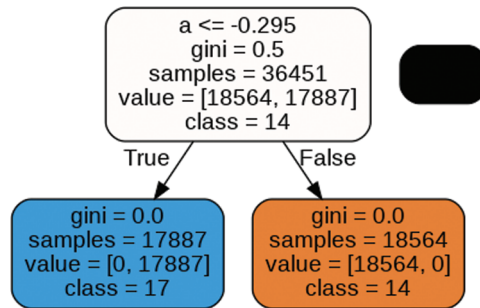


Figure 7: Decision tree with features set

3.4.2 Naïve Bayes

As we are dealing with different classes, the naïve Bayes based on Bayes theorem can help to classify our different classes of hand gestures. So we first trained our model with the naïve Bayes Algorithm. It works best on independent features. It recognizes or predicts on the basis of conditions. For example, if situation B has happened, then what will be the probability or likelihood of happening situation A. The equations below show the Bayes equation on which the classifier works. B is considered an event that has occurred, and A will be the class that will be on the basis of conditions that have occurred, which we call B (angle and line values). Here is the Eq. (13):

$$P\left(\frac{A}{B}\right) = \frac{P\left(\frac{B}{A}\right) P(A)}{P(B)} \tag{13}$$

In Eq. (13) A shows classes and B are the features set. Hence Eq. (13) can be written

$$P\left(\frac{A}{f_1, f_2, f_3, f_4, f_5 \dots f_9}\right) = \frac{P\left(\frac{f_1}{A}\right) \cdot P\left(\frac{f_2}{A}\right) \cdot P\left(\frac{f_3}{A}\right) \cdot P\left(\frac{f_4}{A}\right) \dots \dots \dots P\left(\frac{f_9}{A}\right) \cdot P(A)}{P(f_1) \cdot P(f_2) \cdot P(f_3) \dots \dots \dots P(f_9)} \tag{14}$$

3.4.3 Random Forest

Random Forest or Random Tree Forest classifier consists of many decision trees. It takes multiple random samples from the provided dataset and constructs trees based on those selected features. Then it gets results from different trees that were created over the samples and gives out the final result as an output. When each tree recognizes or generates results, then the majority voting step is done; i.e., the most occurring result voting. Then a final result is selected when the majority voting step is completed. The final result is the most occurring class recognized by different random forest trees. In our proposed method, different hand gestures will be selected randomly from the dataset, and n random answer will be generated as shown in Fig. 8.

3.5 Home Automation

The modeled smart home system in our experiment includes three devices, which are presented in Table 1.

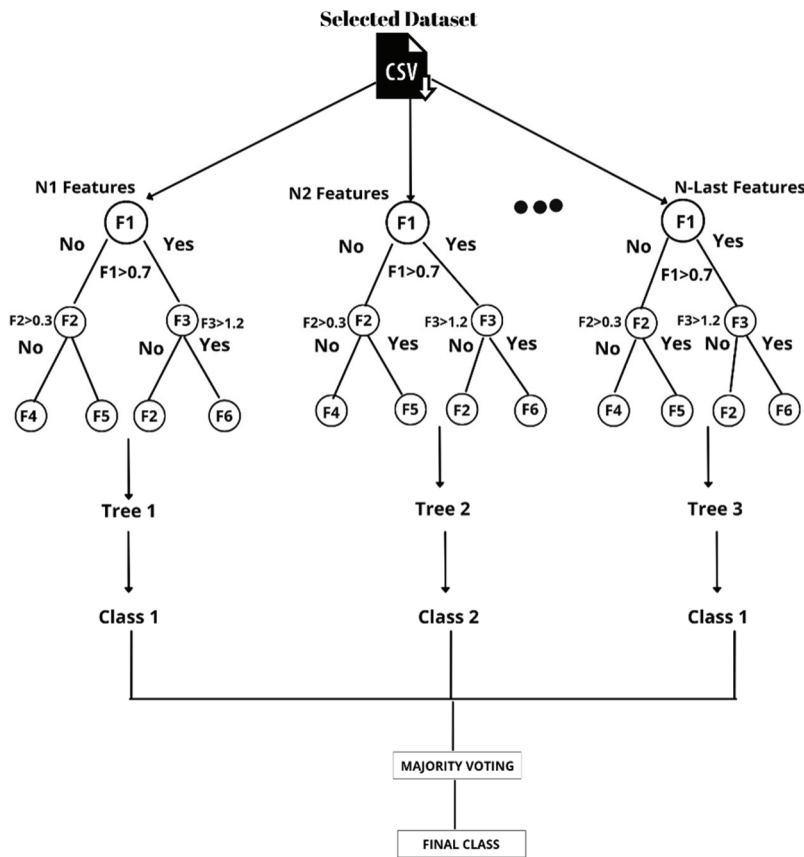


Figure 8: Random forest

Table 1: Devices of the home automation system

Appliance	State variable(SV)	SV allowed values	Wire circuit
Bulb	Power	On off	Relay
Fan	Power	On off	Relay
Door	Movement	Open close	Servo motor

4 Experimental Settings and Results

This section contains a detailed overview of the HGR benchmark dataset. The outcomes of the experiments are then detailed in Section B.

4.1 Dataset Descriptions

As shown in Fig. 9, we proposed a model in which individuals controlled their smart home utilizing 9 gestures. The group of gestures (0, 1, 6, clockwise circle, counterclockwise circle, move down, up, left, and right) is used to alter device states such as on/off, up/down, and mode selection. We also ensure consistent hand gesture recognition, which detects both initial and final points of movements

automatically. Users can start an event by just raising their hand in the air. When the user wants to issue a command, our system will be able to distinguish between unknown and known activities.

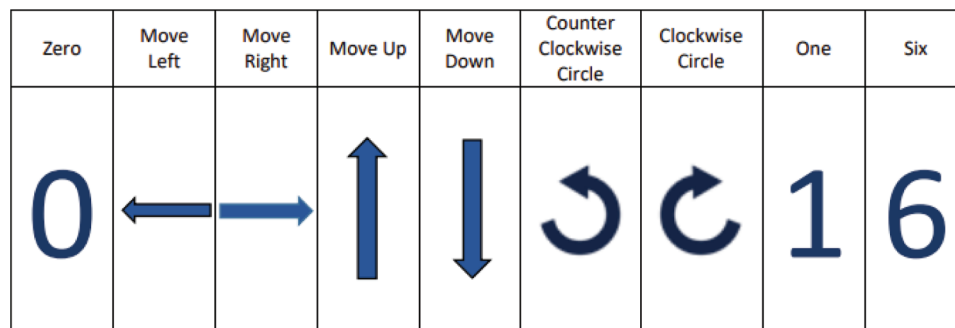


Figure 9: Different gestures vocabulary for HGR

4.2 Experimental Evaluation of HGR Datasets

The proposed system was evaluated on HGR Dataset. In Google Colab, experiments are done using by GPUs and TPUs. The data is split into two sections: 70% for training and 30% for testing. Table 2 displays the HGR dataset results' confusion matrix.

Table 2: Confusion matrix of random forest classifier on hand gesture recognition dataset

Zero	0.50	0.26	0.04	0.01	0.01	0.02	0.02	0.01	0.13
CCW Circle	0.20	0.60	0.07	0.03	0.01	0.01	0.00	0.00	0.09
CW Circle	0.07	0.02	0.78	0.03	0.01	0.02	0.01	0.00	0.07
Move Down	0.00	0.00	0.01	0.99	0.00	0.00	0.00	0.00	0.00
Move Left	0.00	0.00	0.00	0.00	0.89	0.00	0.00	0.00	0.11
Move Right	0.00	0.00	0.00	0.00	0.00	0.90	0.00	0.00	0.10
Move up	0.00	0.00	0.00	0.00	0.00	0.00	0.99	0.00	0.01
One	0.05	0.07	0.12	0.00	0.00	0.02	0.01	0.31	0.43
Six	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.99

Zero CCW Circle CW Circle Move down Move left Move right Move up One Six

Note: CCW = Counter Clockwise Circle, CW = Clockwise Circle.

4.3 Comparison Table

To validate system performance we first selected the HGR (Hand Gesture Recognition) to validate our proposed algorithm. We extracted frames from dataset, applied multiple algorithms, and trained our models. This section also contains the Precision, Accuracy, Recall, and F1 scores of our models over different classifiers. Table 3 shows the Precision, Accuracy, Recall, and F1 scores of our proposed algorithms with the HGR dataset.

Table 3: Accuracy, precision, Recall and F1-score of our proposed classifiers over hand gesture recognition dataset

Dynamic activities	Naïve bayes			Decision tree			Random forest		
	Precision	Re-call	F-measure	Precision	Re-call	F-measure	Precision	Re-call	F-measure
Zero (0)	0.63	0.39	0.48	0.64	0.45	0.53	0.98	0.85	0.87
One (1)	0.48	0.79	0.59	0.54	0.63	0.58	0.85	0.96	0.90
Six (6)	0.61	0.43	0.51	0.48	0.47	0.48	0.94	0.91	0.92
CCW Circle	0.48	0.44	0.46	0.41	0.44	0.42	0.85	0.84	0.85
CW Circle	0.66	0.59	0.63	0.62	0.70	0.66	0.98	0.93	0.96
Move down	0.92	0.97	0.94	0.99	0.96	0.98	1.00	1.00	1.00
Move left	0.87	0.99	0.93	0.90	0.90	0.89	1.00	1.00	1.00
Move right	1.00	0.99	0.99	0.99	0.99	0.99	1.00	1.00	1.00
Move up	1.00	0.99	1.00	1.00	0.99	1.00	1.00	1.00	1.00

Note: CCW = Counter Clockwise, CW = Clockwise.

From [Table 3](#). It is conclusive that the random forest classifier with proposed algorithm is working well as compare to other two classifiers we have selected. The error rate for each classifier is mentioned in the [Table 4](#) for the corresponding classifiers.

Table 4: Error rate calculated for the different classifiers

	Naïve bayes	Decision tree	Random forest
Error rate	0.16	0.14	0.06

5 Conclusion/Summary

We discuss a proposed system for recognizing hand moment gestures using hand glove consist of sensors for the home automation process. Proposed hand gestures discussed in this paper are used to operate appliances in smart-homes that can be easily memorized by the user. In this study, we argue that computer vision techniques are unsatisfactory because they are difficult and expensive, need a proper atmosphere where color is properly calibrated before each usage, and are strongly impacted by light conditions. In comparison with computer vision-based gesture recognition techniques, wearable sensor-based gesture recognition technology is relatively inexpensive, has lower power consumption, needs just moderate computation, requires no color calibration in advance, does not violate users' privacy, and is unaffected by lighting conditions. In this paper, an approach for hand gesture recognition via inertial sensors for home automation is presented to enhance human-machine interactivity. We have achieved an accuracy of 94% over the public benchmarked HGR dataset. The evaluation results revealed that gesture recognition for home automation is feasible. To increase gesture recognition accuracy, future research should look at deep learning models.

Funding Statement: This research was supported by a grant (2021R1F1A1063634) of the Basic Science Research Program through the National Research Foundation (NRF) funded by the Ministry of Education, Republic of Korea.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] J. P. Wachs, H. I. Stern, Y. Edan, M. Gillam, J. Handler *et al.*, “A gesture-based tool for sterile browsing of radiology images,” *Journal of the American Medical Informatics Association*, vol. 15, no. 3, pp. 321–323, 2008.
- [2] P. Gonzalo and A. Juan, “Control of home devices based on hand gestures,” in *Proc. Int. Conf. on Consumer Electronics*, Berlin, pp. 510–514, 2015.
- [3] I. Pavlovic, R. Sharma and T. S. Huang, “Visual interpretation of hand gestures for human-computer interaction: A review,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 677–695, 1997.
- [4] A. Yang, S. Chun and J. Kim, “Detection and recognition of hand gesture for wearable applications in IOMT,” in *20th Int. Conf. on Advanced Communication Technology (ICACT)*, PyeongChang, Korea (South), pp. 1046–1053, 2018.
- [5] C. Pham and N. Thuy, “Real-time traffic activity detection using mobile devices,” in *Proc. of the 10th Int. Conf. on Ubiquitous Information Management and Communication*, Danang, Vietnam, pp. 2016, 2016.
- [6] A. Jalal, B. Mouazma and K. Kim, “Stochastic recognition of physical activity and healthcare using tri-axial inertial wearable sensors,” *Applied Sciences*, vol. 10, no. 20, pp. 1–20, 2020.
- [7] A. Jalal, Q. Majid and A. Hasan, “Wearable sensor-based human behavior understanding and recognition in daily life for smart environments,” in *Proc. of the 10th Int. Conf. on Frontiers of Information Technology*, Islamabad, Pakistan, pp. 1–6, 2018.
- [8] A. Jalal, Q. Majid and M. Sidduqi, “A triaxial acceleration-based human motion detection for ambient smart home system,” in *Proc. of the Int. Bhurban Conf. on Applied Sciences and Technology*, Islamabad, Pakistan, pp. 1–6, 2019.
- [9] A. A. Rafique, A. Jalal and A. Ahmed, “Scene understanding and recognition: Statistical segmented model using geometrical features and Gaussian naïve Bayes,” in *IEEE Conf. on Int. Conf. on Applied and Engineering Mathematics*, Taxila, Pakistan, pp. 225–230, 2019.
- [10] M. Batool, A. Jalal and K. Kim, “Sensors technologies for human activity analysis based on SVM optimized by PSO algorithm,” *IEEE ICAEM Conference*, Taxila, Pakistan, pp. 145–150, 2019.
- [11] M. A. Amin and H. Yan, “Sign language finger alphabet recognition from Gabor-PCA representation of hand gestures,” in *Proc. Int. Conf. on Machine Learning and Cybernetics*, Hong Kong, vol. 4, pp. 2218–2223, 2007.
- [12] S. Sakshi and S. Sukhwinder, “Vision-based hand gesture recognition using deep learning for the interpretation of sign language,” *Expert Systems with Applications*, vol. 182, pp. 115657, 2021.
- [13] N. S. Schack and R. Orngreen, “The effectiveness of e-learning explorative and integrative review of the definitions, methodologies and factors that promote e-learning effectiveness,” *Electronic Journal of E-Learning*, vol. 13, pp. 278–290, 2015.
- [14] K. Yelin, S. Tolga and B. Feyzi, “Towards emotionally aware AI smart classroom: Current issues and directions for engineering and education,” *IEEE Access*, vol. 6, pp. 5308–5331, 2018.
- [15] Z. Zhen, Y. Kuo, Q. Jinwu and Z. Lunwei, “Real-time surface EMG pattern recognition for hand gestures based on an artificial neural network,” *Sensors*, vol. 19, pp. 3170, 2019.
- [16] O. Munir, A. Ali and C. Javaan, “Hand gestures for elderly care using a microsoft kinect,” *Nano Biomedicine and Engineering*, vol. 12, pp. 197–204, 2020.
- [17] P. Francesco, C. Dario and C. Letizia, “Recognition and classification of dynamic hand gestures by a wearable data-glove,” *SN Computer Science*, vol. 2, pp. 1–9, 2021.
- [18] A. Jalal, Y. Kim, Y. Kim, S. Kamal and D. Kim, “Robust human activity recognition from depth video using spatiotemporal multi-fused features,” *Pattern Recognition*, vol. 61, pp. 295–308, 2017.

- [19] A. Jalal, S. Kamal and D. Kim, "A depth video sensor-based life-logging human activity recognition system for elderly care in smart indoor environments," *Sensors*, vol. 14, no. 7, pp. 11735–11759, 2014.
- [20] A. Jalal, M. Z. Uddin and T. Kim, "Depth video-based human activity recognition system using translation and scaling invariant features for life logging at smart home," *IEEE Transactions on Consumer Electronics*, vol. 58, no. 3, pp. 863–871, 2012.
- [21] F. Tsai and K. Chan, "Dimensionality reduction techniques for data exploration," in *Proc. of the Int. Conf. on Information, Communications and Signal Processing*, Singapore, pp. 1–6, 2008.
- [22] F. Song, Z. Guo and D. Mei, "Feature selection using principal component analysis," in *Proc. of the Int. Conf. on System Sciences, Engineering Design and Manufacturing*, Yichang, China, pp. 1–6, 2010.