applied sciences

MDPI

Article

Dynamic Hand Gesture Recognition for Smart Lifecare Routines via K-Ary Tree Hashing Classifier

Hira Ansar [1], Amel Ksibi [2,*], Ahmad Jalal [1], Mohammad Shorfuzzaman [3], Abdulmajeed Alsufyani [3], Suliman A. Alsuhibany [4] and Jeongmin Park [5,*]

[1] Department of Computer Science, Air University, Islamabad 44100, Pakistan; hiraansar53@gmail.com (H.A.); ahmadjalal@mail.au.edu.pk (A.J.)
[2] Information Systems Department, College of Computer and Information Science, Princess Nourah Bint Abdulrahman University, Riyadh 11564, Saudi Arabia
[3] Department of Computer Science, College of Computers and Information Technology, Taif University, Taif 21944, Saudi Arabia; m.shorf@tu.edu.sa (M.S.); a.s.alsufyani@tu.edu.sa (A.A.)
[4] Department of Computer Science, College of Computer, Qassim University, Buraydah 51452, Saudi Arabia; salsuhibany@qu.edu.sa
[5] Department of Computer Engineering, Tech University of Korea, 237 Sangidaehak-ro, Siheung-si 15073, Gyeonggi-do, Korea
* Correspondence: amelksibi@pnu.edu.sa (A.K.); jmpark@tukorea.ac.kr (J.P.)

Featured Application: The proposed system is an image processing module that monitors, tracks, and recognizes hand gestures and has been evaluated over publicly available benchmark datasets. However, this technique can be used over automated home appliances as well as security systems to control surrounding environments and classify their events.

Abstract: In the past few years, home appliances have been influenced by the latest technologies and changes in consumer trends. One of the most desired gadgets of this time is a universal remote control for gestures. Hand gestures are the best way to control home appliances. This paper presents a novel method of recognizing hand gestures for smart home appliances using imaging sensors. The proposed model is divided into six steps. First, preprocessing is done to de-noise the video frames and resize each frame to a specific dimension. Second, the hand is detected using a single shot detector-based convolution neural network (SSD-CNN) model. Third, landmarks are localized on the hand using the skeleton method. Fourth, features are extracted based on point-based trajectories, frame differencing, orientation histograms, and 3D point clouds. Fifth, features are optimized using fuzzy logic, and last, the H-Hash classifier is used for the classification of hand gestures. The system is tested on two benchmark datasets, namely, the IPN hand dataset and Jester dataset. The recognition accuracy on the IPN hand dataset is 88.46% and on Jester datasets is 87.69%. Users can control their smart home appliances, such as television, radio, air conditioner, and vacuum cleaner, using the proposed system.

Keywords: convolution neural network; frame differencing; hand gestures; point-based trajectories; smart home appliances; single shot detector; 3d point clouds; k-ary tree hashing classifier

Citation: Ansar, H.; Ksibi, A.; Jalal, A.; Shorfuzzaman, M.; Alsufyani, A.; Alsuhibany, S.A.; Park, J. Dynamic Hand Gesture Recognition for Smart Lifecare Routines via K-Ary Tree Hashing Classifier. *Appl. Sci.* **2022**, *12*, 6481. https://doi.org/10.3390/app12136481

Academic Editor: Andrea Prati

Received: 25 May 2022
Accepted: 24 June 2022
Published: 26 June 2022

1. Introduction

Over the past few years, intelligent human-computer interaction recognition in the smart home environment is getting more attention in many fields, including architecture, robotics, biomedical, and smart home appliances [1–3]. The easiest way of controlling smart home appliances is through hand gestures [4–6]. The appliances that you use every day are an important part of your home. Consumers today are very careful about their comfort and safety and are more interesting in smart appliances [7–9]. They will facilitate their daily life by controlling the lights, speakers, air conditioners, and similar robots through hand gestures [10–12]. By making gestures, you can control all the devices in your home. A motion sensor is one of the most important tools in your smart home [13–15].

Different models proposed for controlling smart home appliances using hand gesture recognition can be divided into two main streams. The first approach is based on the recognition of hand gestures using motion sensors embedded in smart home appliances [16–18]. In motion-based sensors, one inertial sensor or an array of sensors is used. These sensors are responsible for tracking the acceleration, velocity, and position of the hand. Such type of motion features help to control smart appliances such as television, radio, and lighting of rooms [19–21], however, the drawback of using motion-based sensors in smart home appliances is the high sensitivity. The second approach is the use of image sensors [22–24] or cameras to obtain the commands from hand gestures [25]; the sensors are trained on image features which include color, shape, texture, position contours, and motion of the hands. Our proposed model is based on the second approach and recognizes the hand gestures using imagery sensors or cameras [26].

In this research article, we propose a robust method for recognizing hand gestures for controlling smart home appliances. For this, we use the IPN hand dataset and Jester dataset. Initially, preprocessing of the video samples for frame conversion, motion blur noise reduction, and resizing is performed. The next step is hand detection via SSD-CNN. After that, the hand skeleton is extracted to process these data sources by various algorithms for features extraction, optimization of the extracted features, and recognition of the hand gestures. The main contributions of this paper are:

- For hand motion and position analysis, we propose a method for extracting hand skeletons;
- For the recognition of image-based hand gestures, we have extracted novel features based on point-based trajectories, frame differencing, orientation histogram, and 3D point clouds.

The article is subdivided as follows: we start with the related works section, which is followed by our system methodology. Then, the detailed experimental setup is discussed, and finally, an overview of the paper is presented in the Conclusions Section.

2. Related Work

Hand gesture recognition can help computers translate and interpret specific motions to control smart home appliances. With the advancement of technology, various hand gesture recognition systems have been developed via smart tools and various classification approaches. In this section, we will discuss the detailed description of various HGR models developed in the past few years. Table 1 includes a comprehensive review of recent research in this area.

Table 1. A comprehensive review of relevant research.

Hand Gestures Recognition for Controlling Smart Home Appliances	
Methods	**Main Contributions**
H. Khanh et al. (2019) [24]	The system was developed for controlling smart home appliances using two deep learning models fused with mobile sensors to recognize hand gestures. The mobile sensors were instrumented on smartwatches, smartphones, and smart appliances. The deep learning models helped in the learning and representation of the mobile sensors' data.
Ransalu et al. (2012) [25]	The HGR model was developed to automate the home appliances using hand gestures. First, the hand was detected using the Viola-Jones object detection algorithm. To segment the hand from the image, the YCbCr skin color segmentation technique was used. The filtered hand was refined using dilation and eroding. At last, a multilayer perceptron was used to classify the four-hand gestures, i.e., ready, swing, on/speedup & off.

Table 1. *Cont.*

Hand Gestures Recognition for Controlling Smart Home Appliances	
Methods	**Main Contributions**
P. N. et al. (2017) [26]	The model consists of a few steps. The hand gestures were captured using the web camera and were then preprocessed to detect the hands. Corner point detection was used to de-noise the images using a MATLAB simulation tool. Based on the gestures, different threshold values were used for controlling the home appliances. The threshold values were generated using the Fast Fourier transform algorithm and the appliances were controlled by the micro-controllers.
V. Utpal et al. (2011) [27]	For controlling home appliances and electronic gadgets using hand gestures, the authors detected hands using the YCbCr skin color segmentation model and traced edges. For gesture recognition, the number of fingers was counted, and its orientation was analyzed. The reference background was stored from each frame which was compared with the next frame for reliable hand gesture recognition.
Santhana et al. (2020) [28]	They developed a hand gesture recognition system using Leap motion sensors. The system was customized to recognize multiple motion-based hand gestures for controlling smart home appliances. The system was trained using a customized dataset containing various hand gestures to control daily household devices using a deep neural network.
Qi et al. (2013) [29]	They developed a hand recognition system for controlling television. The system was categorized into three sub-categories. (1) For static hand gesture recognition, hand features were extracted using a histogram-oriented gradient (HOG), and for recognition, Adaboost training was used. (2) For dynamic hand gesture recognition, first the hand trajectory was recognized and passed through the HMM model for recognition. (3) For finger click recognition, a specific depth threshold was fixed to detect the fingers. The distance between the palm and the fingertip was calculated. The accumulated variance was calculated for each fingertip to recognize the finger click gesture.
Yueh et al. (2018) [30]	The authors developed a system to control a TV using hand gestures. First, the hand was detected through skin segmentation and the hand contour was extracted. After that, the system was trained using CNN to recognize hand gestures that were categorized into five branches; (1) menu, (2) direction, (3) go back, (4) mute/unmute, and (5) nothing. After that, CNN also helped in tracking the hand joints to detect commands: (1) increase/ decrease the speed, (2) clicking, and (3) cursor movement.

3. Material and Methods

In this section, a detailed description of the proposed model is given. First, the video input is converted into RGB frames, then the frames are resized to a fixed dimension. Noise is removed and the quality of images is enhanced and sharpened. The second step is hand detection by first removing the background and extracting the foreground. A hand skeleton is extracted for localizing the points on the entire hand. Then the point-based and texture-based features are extracted. These features are optimized by using an optimization algorithm. At last, a classification algorithm is used for the classification of hand gestures for controlling smart home appliances. Figure 1 illustrates the proposed HGR system model structural design.

3.1. Preprocessing of the Input Videos

Before the localization of the hand points, some preprocessing techniques are applied to save computational cost and time. Initially, video data is converted into frames (images). These frames are set to a fixed dimension of 452 × 357. After that, the frames are denoised using the median filtering algorithm. The median filtering is used for detecting the distorted pixels in the images and replacing the corrupted pixels values with the median values. A 5 × 5 window is used to de-noise the image [31–37]. The median filter is defined in Equations (1)–(3);

$$Med\ (I)\ =\ Med\{I_p\} \tag{1}$$

$$= \frac{I_p(k+1)}{2}; k\ is\ odd \tag{2}$$

$$= \frac{1}{2}\left[I_p\left(\frac{k}{2}\right) + I_p\left(\frac{k}{2}\right) + 1\right], \quad (3)$$

where $I_1, I_2, I_3, \ldots, I_k$ is the sequence of neighboring pixels. Before applying the filter, all pixels of the images should be arranged in ascending or descending order. After sorting the pixels, the sequence of the pixels will be $I_{p1} < I_{p2} < I_{p3} < I_{pk}$, where k is usually odd. Figure 2 shows the results of preprocessing on video frames.

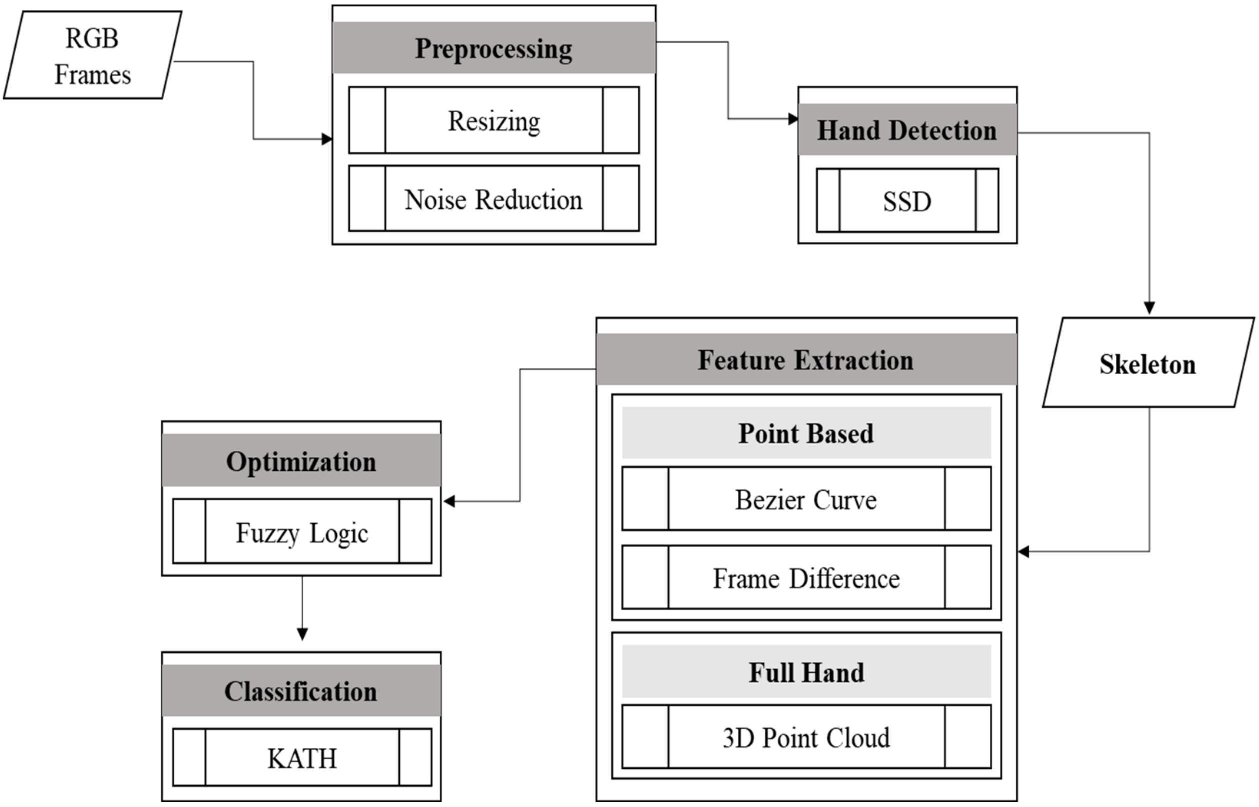

Figure 1. The proposed HGR system model structural design.

Figure 2. Results of preprocessing on video frames over IPN dataset.

3.2. *Hand Detection Using Single Shot MultiBox Detector*

The ridge detection of the human silhouette comprises two steps that are binary edge extraction and ridge data generation [38–40]. In the binary edge extraction step, the binary edges are extracted from the RGB and depth silhouettes are obtained in the described preprocessing stage. The distance maps are produced on the edges by using distance transform (See Figure 3). While, in the ridge data generation step, the local maximal is obtained from the pre-computed maps, which produces ridge data within the binary edges [41,42]. A further description of binary edge detection and generation of ridge data is described below in Algorithm 1:

Algorithm 1: Hand detection using a single shot multibox detector.

Input: Optimized feature vectors.
Output: Hand gesture classification.
Step 1: Check the length of the hash table (say n).
Check the number of entries in the hash table by setting a fixed threshold (say T = 40).
If (n > T) **then**
Find the correlation matching or minimum distance between the vectors by the following equation:

$$\gamma_K = \sum_{x=1}^{n} ||\alpha_x| - |\beta|| \; where \; K = 1,2,3,4$$

where α represents the centroid of the vectors stored in the hash table, β represents the new vectors of the test image, γ represents the distance between the stored values of the hash table and the new vectors.
Now, finding the sum of the vectors

$$\theta = \gamma_1 + \gamma_2 + \gamma_3 + \gamma_4$$

End
Step 2: /*Check the correlation γ_K of the new entry*/
If ($\gamma_K \geq 0.98$)
Match exists
Else
Match does not exist
End

Figure 3. Results of the hand detection using a single-shot multibox detector.

3.3. Hand Landmarks Localization Using Skeleton Method

To localize the hand landmarks points, the first step is to localize the palm region. For doing this, we selected the palm area via a single shot multibox detector and removed the fingers; with this help, a bounding box appears on the palm. Then the extreme top-left, top-right, bottom-left and bottom-right points are calculated and marked with the 4 points [43–46].

The next step is to localize the finger points. For this, the palm region is removed and only the fingers are left. The extreme points are identified with the help of a scanning window that moves from top to bottom, identifying all the extreme points. As a result, the extreme top points of all fingers are marked with the 5 points. Similarly, the extreme bottom points are identified using the scanning window that moves from bottom to top marking 5 points on the bottom of the fingers [47]. Figure 4 shows the results of hand point localization.

3.4. Features Extraction

For features extraction, we have extracted both points-based and appearance-based features for better classification of the hand gestures. For point-based features, we have used Bezier curves and frame differencing. For appearance-based features, 3D point clouds are mapped on the hands.

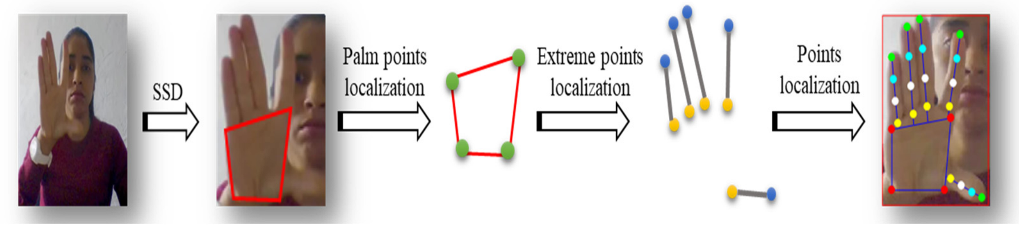

Figure 4. Hand landmarks localization results in hand gestures using the skeleton method.

3.4.1. Bezier Curves

The landmark points localized on the entire hand are utilized for Bezier curves fitting for analyzing the trajectories of the hand in different gestures. For this, we have taken three control points to represent a curve using Equation (4);

$$Curve(x) = \sum_{i=0}^{n} Q_{i,n}(x)Y_i \tag{4}$$

The points along the curve are determined by x, *where* $0 \leq x \leq 1$. The degree of the curve is denoted by n, which is one less than the control points. Y_i is the *i-th* control point where $Y(0) = Curve(0)$ and $Y(n) = Curve(1)$. $Q_{i,n}$ is the Bernstein polynomial and is calculated in Equation (5).

$$Q_{i,n} = \frac{n!}{i!(n-i)!} n^i (1-x)^{n-i} \tag{5}$$

We have used Bezier curves with three control points describing the quadratic curve. Therefore, the Bernstein polynomial with $n = 2$ is calculated as in Equations (6)–(8) [48–51].

$$Q_{0,2} = (1-x)^2 \tag{6}$$

$$Q_{1,2} = 2x(1-x) \tag{7}$$

$$Q_{2,2} = x^2 \tag{8}$$

Therefore, the equation of a Bezier curve with three control points is simplified as in Equation (9). Figure 5 shows the results of the Bezier curves fitting on the hand.

$$Curve(x) = (1-x)^2 + 2x(1-x) + x^2 \tag{9}$$

Figure 5. The results of the Bezier curve.

3.4.2. Frame Differencing

Keyframes are the representation of the elements in the image sequences. In this model, the keyframes have been extracted that exhibit the dynamic hand gestures. Each hand shows a gesture that is localized by a set of points. To find the difference in the positions of the landmarks, the first and the pause frame sequences are taken as the keyframes. Figure 6 illustrates the change in the position of the pixel between different frames. The first frame is well established. To find the motion of the hand gestures, we have adopted the following method of frame difference as defined in Equation (10) [52–56].

$$Diff_k(x,y) = |Frame_k(x,y) - Frame_{k-1}(x,y)| \quad (10)$$

where $Frame_k(x,y)$ and $Frame_{k-1}(x,y)$ are the two consecutive frames in which the hand is not moving and $Diff_k$ is defined as in Equation (11).

$$Diff_k(x,y) = |Frame_k(x,y) - Frame_{k-1}(x,y)| \approx 0 \quad (11)$$

where $Frame_k(x,y)$ is the pause keyframe. The two continuous frames are impossible to be the keyframes. Thus, if $Frame_{k-1}(x,y)$ is the keyframe, then $Frame_k(x,y)$ is not the keyframe. Therefore, for each frame sequence, $Frame_k(x,y)$ with N number of frames: The following approach should be outlined as:

1. Initialize the keyframe number with $n = 1$. So, the keyframes are marked as $M_1(x) = 1$, $M_k(x) = 0$, $k = (2,\ldots, N)$. After that, compute the difference frame $Diff_k$ between $Frame_k(x,y)$ and $Frame_{k-1}(x,y)$, $k = (2,\ldots, N)$. Compute the valid pixels of N.
2. If $N > Thresh_1$ and $M_{k-1}(x) = 0$, set $n = n + 1$, set $M_k(x) = 1$.
3. Set $k = k + 1$ if the value of k is less than N. Then repeat the steps, otherwise end the procedure.
4. After calculating the frame difference, each key point L in the first keyframe and L' points in the other keyframes positions are calculated using the distance formula defined in Equation (12).

$$Distance = \sqrt{(L1' - L1)^2 + (L2' - L2)} \quad (12)$$

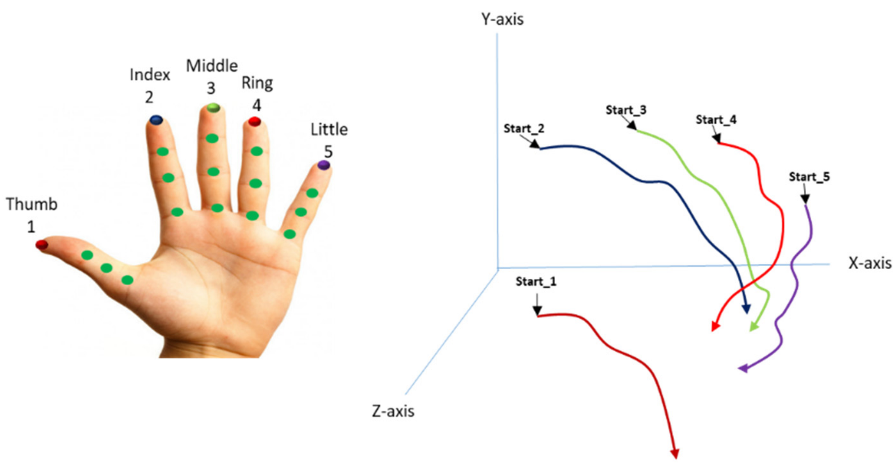

Figure 6. Illustration of the change in the direction between different frames' extreme points.

D Point Clouds

For appearance-based features, extracted the 3D point clouds. The following are the main steps of extracting the appearance-based features [57–59].

1. First, a central point on the palm is taken and the maximum distance d between the central point and the edges E_i point of the gesture region is calculated. After that, ten different lengths of the radius are defined as $X = n \times \frac{d}{10}$ $where\ n = 1, 2, 3, \ldots 10$. Next, the center of the rhombus is defined as C and the radiuses as r_n. We have drawn 10 rhombuses (innermost is the first rhombus and outmost is the tenth rhombus) as shown in Figure 7. To highlight the effect of changing hand gestures, the color of the rhombus changes on the hand.
2. In Figure 7, it is visible that every rhombus has a different number of intersections with hand gesture regions. For finding the number of stretched fingers S, we have

taken the sixth rhombus (according to the thumb rule). In the sixth rhombus, we extracted those points whose colors vary from green to yellow and yellow to green. We define G_i as the point whose color changes from green to yellow and Y_i as the point whose values changes from yellow to green.

3. Now, for midpoint identification, we define it as M_i which is the midpoint of G_i and Y_i. Then, each midpoint M_i and the central point C can be connected through a line and the angles between the adjacent lines are calculated. The angles are represented as An_j ($j = 1, 2, 3 \ldots\ I - 1$).
4. Using the thumb rule, the fifth rhombus is taken as a boundary line to divide the hand gesture into two parts. For instance, we have taken the first part as $P1$ and the second part as $P2$, where $P1$ lies inside the rhombus and $P2$ is the outside area of the rhombus. Then the ratio R of $P1$ and $P2$ is calculated. The R is the gesture region area distribution feature as shown in Algorithm 2. Figure 7 shows the appearance features using 3D point clouds.

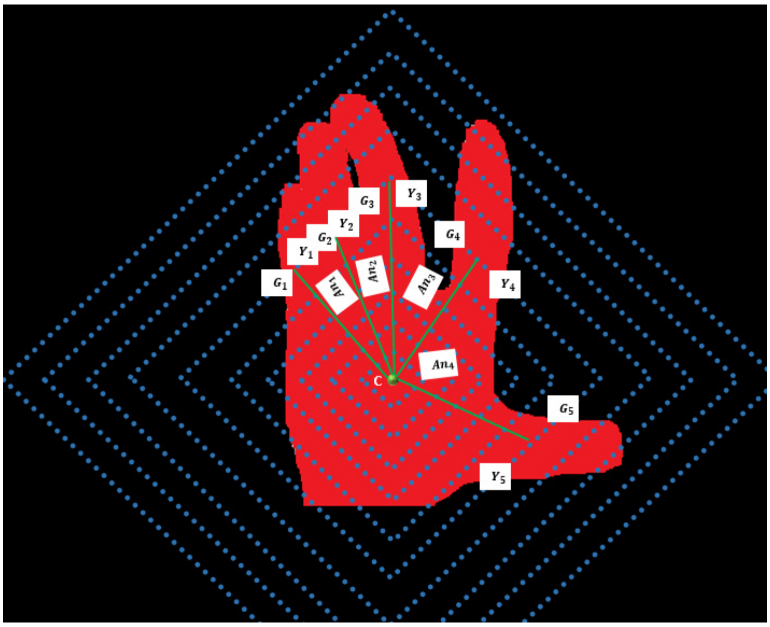

Figure 7. The appearance features using 3D point clouds.

Algorithm 2: Feature extraction.

Input: Hands Point based and texture-based data (x, y, z).
Output: Feature Vectors (v_1, v_2, . . . , v_n).
featureVectors ← [ ]
window_size ← Get windowSize( )
Overlap ← Get Overlapping Time( )
For HandComponent in [x,y,z] do
Hand←window ← Getwindow(HandComponent)
/* Extracting features */
BezierCurves ← ExtractBezierCurvesFeatures (Hand_window)
Frame Differencing ← ExtractFrameDifferencingFeatures (Hand_window)
3D Point Clouds ← Extract3DPointCloudsFeatures (Hand_window)
featureVectors ← GetFeatureVectors [BezierCurves, Frame Differencing, 3D Point Clouds]
featureVectors.append (featureVectors)
End for
featureVectors ← Normalize (featureVectors)
return *featureVectors*

3.5. Feature FOptimization Using Fuzzy Logic

The objective of using fuzzy logic optimizer is to recognize the hand gestures based on the information obtained through different feature descriptors. Each feature descriptor value is labeled with a specific variable and is mapped to their respective fuzzy sets. For instance, we have five fingers labeled as; F1 (thumb), F2 (index finger), F3 (middle finger), F4 (ring finger), and F5 (little finger). The joints of the fingers are labeled as J1, J2, J3, and J4. Similarly, the distance between the fingers is denoted by Di,j showing the distance between the fingers Fi and Fj.

Since any movement of the hand shows the variation of the position of the hand in a sequence of images (frames), to simulate the data transfer, a hand configuration is generated by a tuple of angles. For each tuple of angles, the data is represented using a set of linguistic variables such as curve, straight, and bent. The separation of the fingers is represented as open, closed, crossed, and semi-open. By these notations, the set of features is optimized, helping to reduce the overall computational time and complexity [60–63]. Figure 8 shows the result of the fuzzy logic optimization.

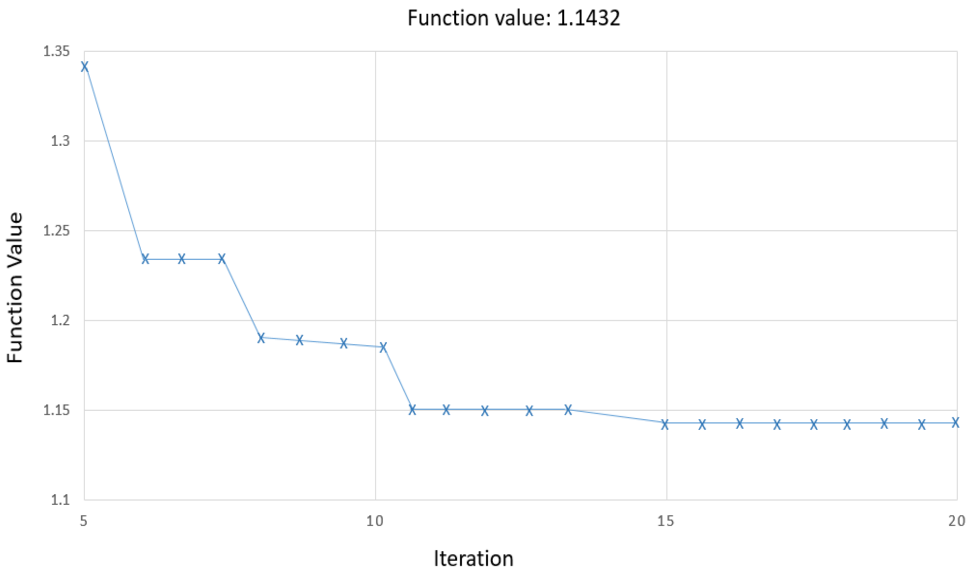

Figure 8. The result of the fuzzy logic optimization.

3.6. Hand Gestures Recognition

For hand gesture recognition, K-Ary Tree Hashing (KATH) classifier is used for the first time in our proposed model [64–67]. The KATH classifier takes the feature descriptors values of each image corresponding to the hand gesture and projects in a common space without the subtree pattern prior knowledge. Then similar pattern feature descriptors are kept in the traversal table. The unique patterns are specified by passing through recursive indexing N numbers to generate $(n-1)$. After that, the hand gesture is classified by the sub-patterns created by MinHash. The experimental results show that KATH classified different hand gestures more accurately than many other state-of-the-art methods, i.e., ANN and decision tree, as shown in Figure 9. In our proposed model, the graph g = (v, ε, l) is given input with the number of iterations I and F representing the feature space. To assign a new label l, the nodes v are relabeled considering the neighboring nodes WV. The traversal table T is generated and stored. After the traversal table, MinHash classifies data. For dimensional reduction, PCA is used to plot results in 3D feature space.

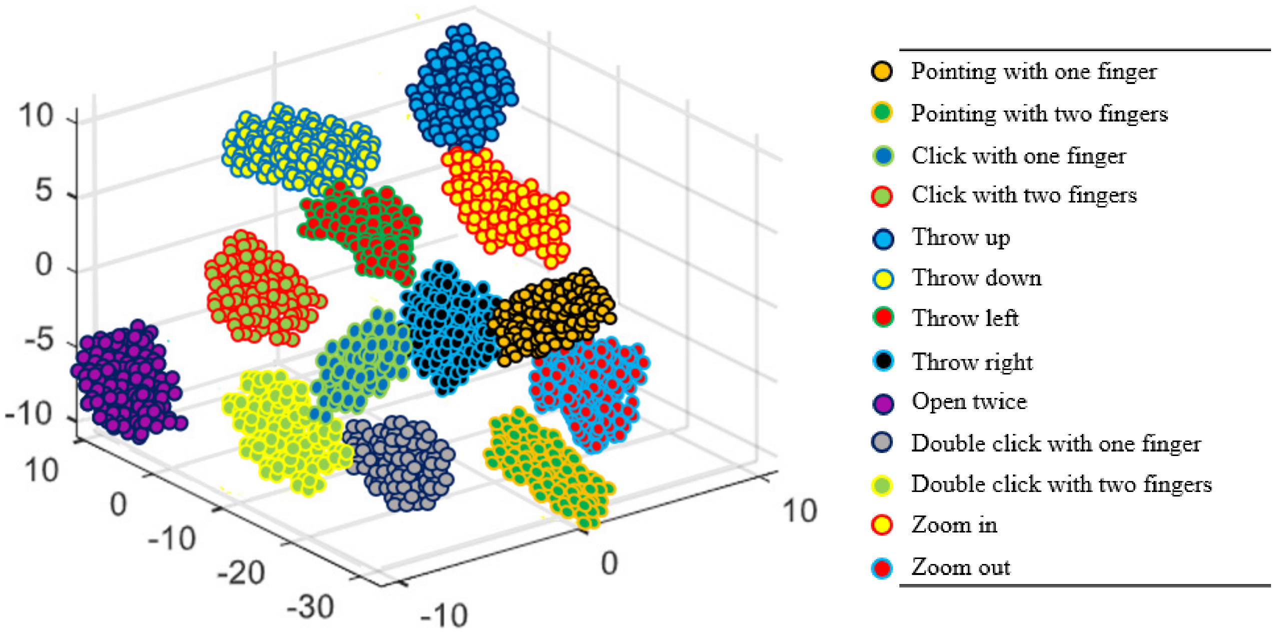

Figure 9. KATH classifier of optimized data over IPN hand dataset for our proposed model.

4. Experimental Setting and Results

4.1. Datasets Descriptions

The IPN hand dataset [68] is a large-scale hand gesture video dataset. It contains 13 gestures, including pointing with one finger, pointing with two fingers, click with one finger, click with two fingers, throw up, throw down, throw left, throw right, open twice, double click with one finger, click with two fingers, zoom in and zoom out. The IPN dataset contains RGB videos with a resolution of 640 × 480 at 30 fps. Figure 10 shows the example images of the IPN hand dataset.

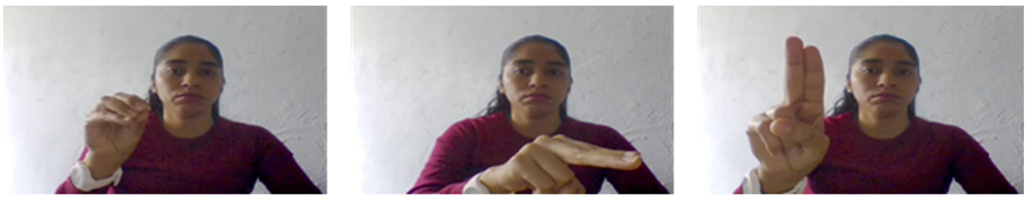

Figure 10. A few example images of the IPN Hand dataset.

The Jester dataset [69] contains a large collection of labeled hand gestures video clips collected by webcam. The dataset contains 148,092 videos, and each video frame is converted into a jpg image at the rate of 12 frames per second. There are 27 classes of hand gestures named: swiping down, swiping left, swiping right, swiping up, thumb down, thumb up, zooming in with full hand, zooming out with full hand, stop, and so on. Figure 11 shows the example images of the Jester dataset.

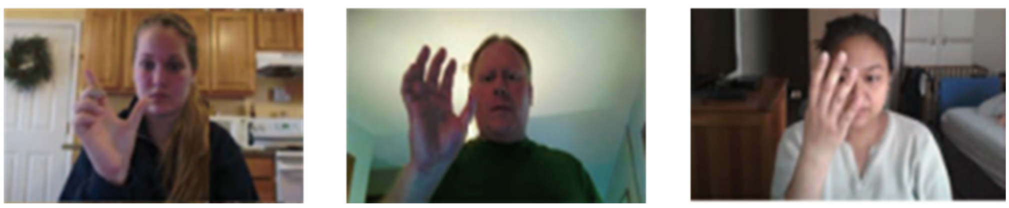

Figure 11. A few example images of the IPN Hand dataset.

4.2. Performance Parameters and Evaluations

4.2.1. Experiment I: The Hand Detection Accuracies

In this experiment, the hand detection accuracies on different hand gestures over IPN Hand dataset and Jester dataset are shown in Tables 2 and 3, respectively. Table 2 represents the results of the hand gestures of the IPN Hand dataset on both plain and complex backgrounds. We took 30 samples of each hand gesture in the plain and complex background and obtained 97.1% accuracy on plain background samples and 94.3% accuracy on complex background samples.

Table 2. Hand detection accuracies over IPN Hand dataset.

Hand Gestures	Number of Samples	Plain Background	Accuracy (%)	Cluttered Background	Accuracy (%)
POF	30	30	100	25	83.3
PTF	30	30	100	26	86.6
COF	30	30	100	26	86.6
CTF	30	28	93.3	26	86.6
TU	30	29	96.6	29	96.6
TD	30	29	96.6	30	100
TL	30	27	90	30	100
TR	30	28	93.3	30	100
OT	30	29	96.6	30	100
DCOF	30	30	100	29	96.6
DCTF	30	30	100	29	96.6
ZI	30	30	100	28	93.3
ZO	30	29	96.6	30	100
Mean Accuracy Rate			**97.1%**		**94.3%**

POF = pointing with one finger, PTF = pointing with two fingers, COF = click with one finger, CTF = click with two fingers, TU = throw up, TD = throw down, TL = throw left, TR = throw right, OT = open twice, DCOF = double click with one finger, DCTF = double click with two fingers, ZI = zoom in, ZO = zoom out.

Table 3. Hand detection accuracies over Jester Hand dataset.

Hand Gestures	Number of Samples	Plain Background	Accuracy (%)	Cluttered Background	Accuracy (%)
SD	30	30	100	25	83.3
SL	30	29	96.6	27	90
SR	30	28	93.3	24	80
SU	30	28	93.3	26	86.6
TD	30	29	96.6	24	80
TU	30	28	93.3	30	100
ZIF	30	29	96.6	25	83.3
ZOF	30	27	90	30	100
S	30	29	96.6	25	83.3
RF	30	30	100	29	96.6
RB	30	30	100	23	76.6
PI	30	28	93.3	28	93.3
SH	30	29	96.6	30	100
Mean Accuracy Rate			**95.6%**		**88.6%**

SD = swiping down, SL = swiping left, SR = swiping right, SU = swiping up, TD = thumb down, TU = thumb up, ZIF = zooming in with full hand, ZOF = zooming out with full hand, S = stop, RF=rolling hand forward, RB = rolling hand backward, PI = pulling hand in, SH = shaking hand.

Table 3 represents the results of the 13 hand gestures of the Jester dataset on both plain and complex backgrounds. We took 30 samples of each hand gesture in the plain and complex background and obtained 95.6% accuracy on plain background samples and 88.6% accuracy on complex background samples.

4.2.2. Experiment II: Hand Gestures Classification Accuracies

For hand gestures classification, we used a KATH classifier. The design method was evaluated using the leave one subject out (LOSO) cross-validation method. In Table 4, the results over the IPN hand video dataset show 88.46% hand gestures classification accuracy. Table 5 represents the confusion matrix for the Jester dataset with 87.69% mean accuracy for hand gestures classification.

Table 4. Confusion Matrix results over IPN Hand dataset.

Class	POF	PTF	COF	CTF	TU	TD	TL	TR	OT	DCOF	DCTF	ZI	ZO
POF	8	1	0	1	0	0	0	0	0	0	0	0	0
PTF	0	9	0	0	0	0	1	0	0	0	0	0	0
COF	0	0	10	0	0	0	0	0	0	0	0	0	0
CTF	0	1	0	9	0	0	0	0	0	0	0	0	0
TU	1	0	0	0	9	0	0	0	0	0	0	0	0
TD	0	0	1	0	0	8	0	0	0	1	0	0	0
TL	0	2	0	0	0	0	7	0	0	0	0	1	0
TR	1	0	0	0	0	0	0	8	0	0	0	1	0
OT	0	0	0	0	0	0	0	0	9	0	1	0	0
DCOF	0	0	0	0	0	0	0	0	0	10	0	0	0
DCTF	0	0	0	0	0	0	0	0	0	0	9	0	1
ZI	0	0	0	0	0	0	0	0	0	0	0	10	0
ZO	0	0	0	0	0	0	0	0	0	0	0	1	9
Hand Gestures classification mean accuracy = 88.46%													

POF = pointing with one finger, PTF = pointing with two fingers, COF = click with one finger, CTF = click with two fingers, TU = throw up, TD = throw down, TL = throw left, TR = throw right, OT = open twice, DCOF = double click with one finger, DCTF = double click with two fingers, ZI = zoom in, ZO= zoom out.

Table 5. Confusion Matrix results over Jester dataset.

Gestures	SD	SL	SR	SU	TD	TU	ZIF	ZOF	S	RF	RB	PI	SH
SD	9	0	0	0	1	0	0	0	0	0	0	0	0
SL	0	9	0	0	0	0	0	0	1	0	0	0	0
SR	0	0	9	0	1	0	0	0	0	0	0	0	0
SU	0	0	1	8	1	0	0	0	0	0	0	0	0
TD	0	0	0	0	10	0	0	0	0	0	0	0	0
TU	0	0	0	0	1	8	0	0	1	0	0	0	0
ZIF	0	0	0	0	0	0	9	0	0	1	0	0	0
ZOF	0	0	0	0	1	0	0	9	0	0	0	0	0
S	0	1	0	0	0	0	0	0	8	0	0	1	0
RF	0	0	0	0	1	0	0	0	0	9	0	0	0
RB	0	0	0	0	0	0	0	0	0	0	10	0	0
PI	0	0	0	0	0	0	0	0	0	0	1	9	0
SH	0	0	0	0	0	0	1	0	0	0	1	1	7
Hand Gestures classification mean accuracy = 87.69%													

SD = swiping down, SL = swiping left, SR = swiping right, SU = swiping up, TD = thumb down, TU = thumb up, ZIF = zooming in with full hand, ZOF = zooming out with full hand, S = stop, RF = rolling hand forward, RB = rolling hand backward, PI = pulling hand in, SH = shaking hand.

4.2.3. Experiment III: Comparison with Other Classification Algorithms

In this segment, we compared the recall, precision, and F1-measure over the IPN hand dataset and the Jester dataset. For the classification of hand gestures, we used a decision tree, an artificial neural network, and we associated the consequences with the KATH classifier. Figure 12 shows the results over the IPN hand dataset and Figure 13 shows the results over the Jester dataset.

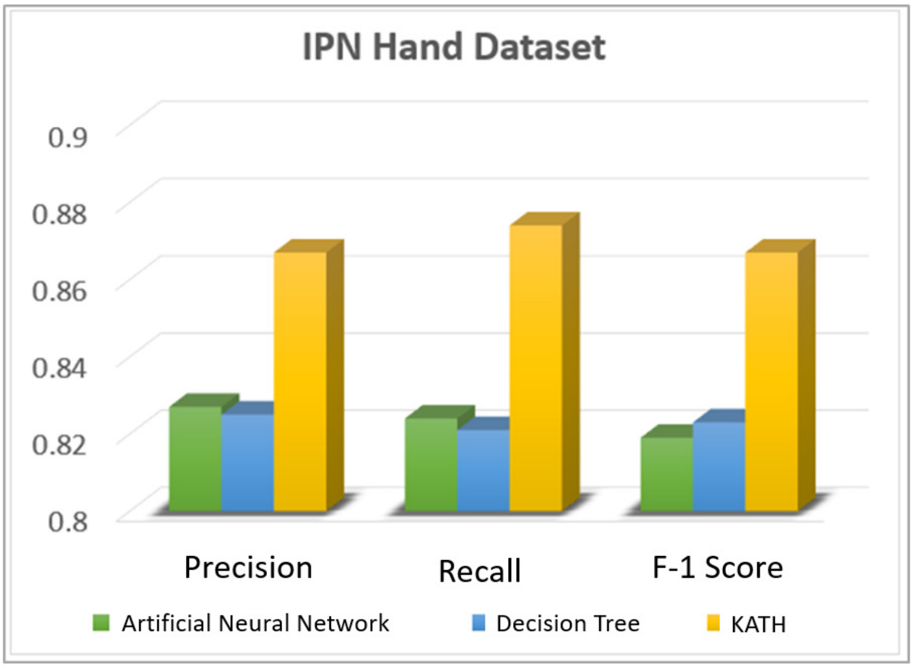

Figure 12. Comparison results of Precision, recall, and F-1 Score using different classifiers over IPN Hand dataset.

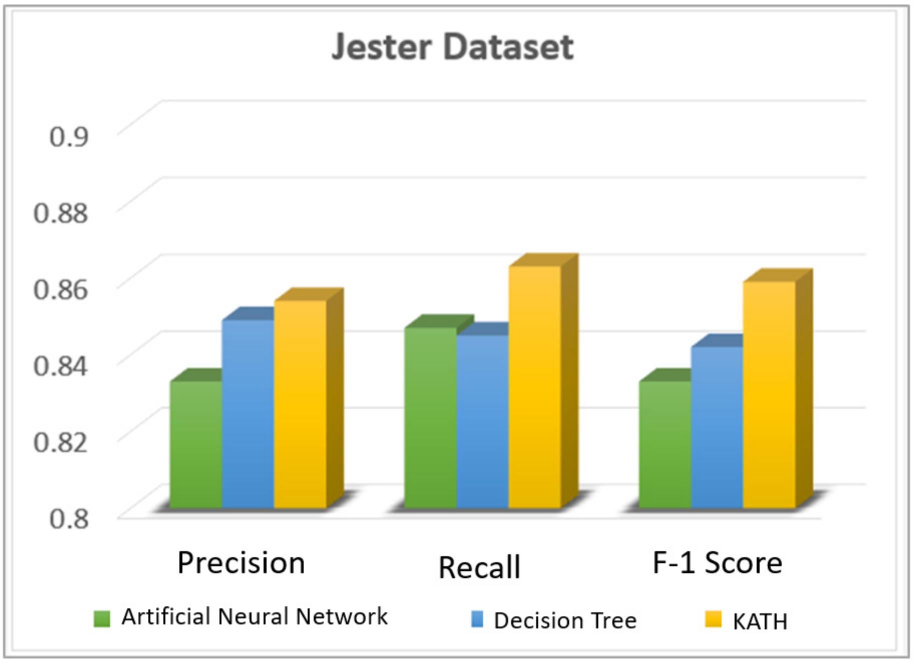

Figure 13. Comparison results of precision, recall, and F-1 Score using different classifiers over the Jester dataset.

4.2.4. Experiment IV: Comparison of our Proposed System with State-of-the-Art Techniques

In this section, we have compared the proposed model with other well-known techniques using the same datasets. Table 6 shows the comparative results between the proposed model and other state-of-the-art techniques.

Table 6. Hand gestures recognition results from the proposed model with other state-of-the-art techniques.

Authors	IPN Hand Dataset (%)	Authors	Jester Dataset (%)
Yamaguchi et al. (2022) [70]	60.00	Zhou et al. (2018) [71]	82.02
Gammulle et al. (2021) [72]	80.03	Shi et al. (2019) [73]	82.34
Garcia et al. (2020) [68]	82.36	Kopuklu et al. (2018) [74]	84.70
TSN [75]	68.01	MFFs [76]	84.70
Proposed method	**88.46**	**Proposed method**	**87.69**

5. Conclusions

This article is based on a hand gestures recognition system for controlling smart home appliances. Two benchmark datasets were selected for experiments: the IPN hand dataset and Jester dataset. Initially, images are acquired, in which hands are detected and landmarks are localized on the palm and the fingers. After that, the textures-based and point-based features are extracted. The hand skeletons are used for extracting the point-based features, whereas the full hand is used for extracting the texture-based features. For feature reduction and optimization, the fuzzy logic is adopted, and finally, the K-ary classification algorithm is used for classifying the hand gestures for operating smart home appliances. For the IPN hand dataset, we achieved the mean accuracy of 88.46% and for the Jester dataset, a mean accuracy of 87.69% was achieved. The proposed system's performance shows a significant improvement compared to existing state-of-the-art frameworks. The limitation of the proposed framework is due to the complexity in the videos, such as cluttered backgrounds and various illumination conditions, which make it difficult to achieve more accurate results.

Author Contributions: Conceptualization, H.A. and A.K.; methodology, H.A. and A.J.; software, H.A. and A.K.; validation, H.A., M.S., S.A.A. and A.A.; formal analysis, A.K., S.A.A. and J.P.; resources, M.S., A.K., A.A. and J.P.; writing—review and editing, H.A., A.K., S.A.A. and J.P.; funding acquisition, A.A., S.A.A. and J.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2023-2018-0-01426) supervised by the IITP (Institute for Information & Communications Technology Planning and Evaluation). In addition, the authors would like to thank the support of the Deanship of Scientific Research at Princess Nourah bint Abdulrahman University. This work was partially supported by the Taif University Researchers Supporting Project number (TURSP-2020/115), Taif University, Taif, Saudi Arabia.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Khan, M.A.; Javed, K.; Khan, S.A.; Saba, T.; Habib, U.; Khan, J.A.; Abbasi, A.A. Human action recognition using fusion of multiview and deep features: An application to video surveillance. *Multimed. Tools Appl.* **2020**, *19*, 1–27. [CrossRef]
2. Zou, Y.; Shi, Y.; Shi, D.; Wang, Y.; Liang, Y.; Tian, Y. Adaptation-Oriented Feature Projection for One-shot Action Recognition. *IEEE Trans. Multimed.* **2020**, *99*, 10. [CrossRef]
3. Ghadi, Y.; Akhter, I.; Alarfaj, M.; Jalal, A.; Kim, K. Syntactic model-based human body 3D reconstruction and event classification via association based features mining and deep learning. *PeerJ Comput. Sci.* **2021**, *7*, e764. [CrossRef] [PubMed]
4. Van der Kruk, E.; Reijne, M.M. Accuracy of human motion capture systems for sport applications; state-of-the-art review. *Eur. J. Sport Sci.* **2018**, *18*, 6. [CrossRef] [PubMed]
5. Wang, Y.; Mori, G. Multiple tree models for occlusion and spatial constraints in human pose estimation. In Proceedings of the European Conference on Computer Vision, Marseille, France, 12–18 October 2008; Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Springer: Berlin/Heidelberg, Germany, 2008.
6. Amft, O.; Tröster, G. Recognition of dietary activity events using on-body sensors. *Artif. Intell. Med.* **2008**, *42*, 121–136. [CrossRef] [PubMed]

7. Sun, S.; Kuang, Z.; Sheng, L.; Ouyang, W.; Zhang, W. Optical Flow Guided Feature: A Fast and Robust Motion Representation for Video Action Recognition. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018.
8. Zhu, Y.; Zhou, K.; Wang, M.; Zhao, Y.; Zhao, Z. A comprehensive solution for detecting events in complex surveillance videos. *Multimed. Tools Appl.* **2019**, *78*, 1. [CrossRef]
9. Akhter, I.; Jalal, A.; Kim, K. Adaptive Pose Estimation for Gait Event Detection Using Context-Aware Model and Hierarchical Optimization. *J. Electr. Eng. Technol.* **2021**, *16*, 2721–2729. [CrossRef]
10. Jalal, A.; Lee, S.; Kim, J.T.; Kim, T.S. Human activity recognition via the features of labeled depth body parts. In Proceedings of the International Conference on Smart Homes and Health Telematics, Artiminio, Italy, 12–15 June 2012; Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Springer: Berlin/Heidelberg, Germany, 2012.
11. Ghadi, Y.; Manahil, W.; Tamara, S.; Suliman, A.; Jalal, A.; Park, J. Automated parts-based model for recognizing human-object interactions from aerial imagery with fully convolutional network. *Remote Sens.* **2022**, *14*, 1492. [CrossRef]
12. Jalal, A.; Sarif, N.; Kim, J.T.; Kim, T.S. Human activity recognition via recognized body parts of human depth silhouettes for residents monitoring services at smart home. *Indoor Built Environ.* **2013**, *22*, 271–279. [CrossRef]
13. Jalal, A.; Kim, Y.; Kim, D. Ridge body parts features for human pose estimation and recognition from RGB-D video data. In Proceedings of the Fifth International Conference on Computing, Communications and Networking Technologies (ICCCNT), Hefei, China, 11–14 July 2014; pp. 1–6.
14. Akhter, I.; Jalal, A.; Kim, K. Pose estimation and detection for event recognition using Sense-Aware features and Adaboost classifier. In Proceedings of the 2021 International Bhurban Conference on Applied Sciences and Technologies (IBCAST), Islamabad, Pakistan, 12–16 January 2021.
15. Jalal, A.; Kamal, S.; Kim, D. Depth Map-based Human Activity Tracking and Recognition Using Body Joints Features and Self-Organized Map. In Proceedings of the 5th International Conference on Computing, Communications and Networking Technologies (ICCCNT), Hefei, China, 11–13 July 2014.
16. Ghadi, Y.; Akhter, I.; Suliman, A.; Tamara, S.; Jalal, A.; Park, J. Multiple events detection using context-intelligence features. *IASC* **2022**, *34*, 3.
17. Jalal, A.; Kamal, S. Real-time life logging via a depth silhouette-based human activity recognition system for smart home services. In Proceedings of the 2014 11th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), Seoul, Korea, 26–29 August 2014; pp. 74–80.
18. Jalal, A.; Kamal, S.; Kim, D. A depth video sensor-based life-logging human activity recognition system for elderly care in smart indoor environments. *Sensors* **2014**, *14*, 11735–11759. [CrossRef] [PubMed]
19. Ghadi, Y.Y.; Akhter, I.; Aljuaid, H.; Gochoo, M.; Alsuhibany, S.A.; Jalal, A.; Park, J. Extrinsic Behavior Prediction of Pedestrians via Maximum Entropy Markov Model and Graph-Based Features Mining. *Appl. Sci.* **2022**, *12*, 5985. [CrossRef]
20. Gochoo, M.; Tahir, S.B.U.D.; Jalal, A.; Kim, K. Monitoring Real-Time Personal Locomotion Behaviors Over Smart Indoor-Outdoor Environments Via Body-Worn Sensors. *IEEE Access* **2021**, *9*, 70556–70570. [CrossRef]
21. Pervaiz, M.; Ghadi, Y.Y.; Gochoo, M.; Jalal, A.; Kamal, S.; Kim, D.-S. A Smart Surveillance System for People Counting and Tracking Using Particle Flow and Modified SOM. *Sustainability* **2021**, *13*, 5367. [CrossRef]
22. Jalal, A.; Akhtar, I.; Kim, K. Human Posture Estimation and Sustainable Events Classification via Pseudo-2D Stick Model and K-ary Tree Hashing. *Sustainability* **2020**, *12*, 9814. [CrossRef]
23. Khalid, N.; Ghadi, Y.Y.; Gochoo, M.; Jalal, A.; Kim, K. Semantic Recognition of Human-Object Interactions via Gaussian-Based Elliptical Modeling and Pixel-Level Labeling. *IEEE Access* **2021**, *9*, 111249–111266. [CrossRef]
24. Trong, K.N.; Bui, H.; Pham, C. Recognizing hand gestures for controlling home appliances with mobile sensors. In Proceedings of the 2019 11th International Conference on Knowledge and Systems Engineering (KSE), Da Nang, Vietnam, 24–26 October 2019; pp. 1–7.
25. Senanayake, R.; Kumarawadu, S. A robust vision-based hand gesture recognition system for appliance control in smart homes. In Proceedings of the 2012 IEEE International Conference on Signal Processing, Communication and Computing (ICSPCC 2012), Hong Kong, China, 12–15 August 2012; pp. 760–763.
26. Chong, Y.; Huang, J.; Pan, S. Hand Gesture recognition using appearance features based on 3D point cloud. *J. Softw. Eng. Appl.* **2016**, *9*, 103–111. [CrossRef]
27. Solanki, U.V.; Desai, N.H. Hand gesture based remote control for home appliances: Handmote. In Proceedings of the 2011 World Congress on Information and Communication Technologies, Mumbai, India, 11–14 December 2011; pp. 419–423.
28. Jamaludin, N.A.N.; Fang, O.H. Dynamic Hand Gesture to Text using Leap Motion. *Int. J. Adv. Comput. Sci. Appl.* **2019**, *10*, 199–204. [CrossRef]
29. Chellaswamy, C.; Durgadevi, J.J.; Srinivasan, S. An intelligent hand gesture recognition system using fuzzy logic. In Proceedings of the IET Chennai Fourth International Conference on Sustainable Energy and Intelligent Systems (SEISCON 2013), Chennai, India, 12–14 December 2013; pp. 326–332.
30. Yang, P.-Y.; Ho, K.-H.; Chen, H.-C.; Chien, M.-Y. Exercise training improves sleep quality in middle-aged and older adults with sleep problems: A systematic review. *J. Physiother.* **2012**, *58*, 157–163. [CrossRef]

31. Farooq, A.; Jalal, A.; Kamal, S. Dense RGB-D Map-Based Human Tracking and Activity Recognition using Skin Joints Features and Self-Organizing Map. *KSII Trans. Internet Inf. Syst.* **2015**, *9*, 1856–1869.
32. Jalal, A.; Kamal, S.; Kim, D. Depth silhouettes context: A new robust feature for human tracking and activity recognition based on embedded HMMs. In Proceedings of the 2015 12th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI 2015), Goyang City, Korea, 28 October 2015.
33. Jalal, A.; Kamal, S.; Kim, D. Individual detection-tracking-recognition using depth activity images. In Proceedings of the 2015 12th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), Goyangi, Korea, 28–30 October 2015; pp. 450–455.
34. Kamal, S.; Jalal, A. A hybrid feature extraction approach for human detection, tracking and activity recognition using depth sensors. *Arab. J. Sci. Eng.* **2016**, *41*, 1043–1051. [CrossRef]
35. Jalal, A.; Kim, Y.-H.; Kim, Y.-J.; Kamal, S.; Kim, D. Robust human activity recognition from depth video using spatiotemporal multi-fused features. *Pattern Recognit.* **2017**, *61*, 295–308. [CrossRef]
36. Kamal, S.; Jalal, A.; Kim, D. Depth images-based human detection, tracking and activity recognition using spatiotemporal features and modified HMM. *J. Electr. Eng. Technol.* **2016**, *11*, 1857–1862. [CrossRef]
37. Gochoo, M.; Akhter, I.; Jalal, A.; Kim, K. Stochastic Remote Sensing Event Classification over Adaptive Posture Estimation via Multifused Data and Deep Belief Network. *Remote Sens.* **2021**, *13*, 912. [CrossRef]
38. Jalal, A.; Kamal, S.; Kim, D. Facial Expression recognition using 1D transform features and Hidden Markov Model. *J. Electr. Eng. Technol.* **2017**, *12*, 1657–1662.
39. Jalal, A.; Kamal, S.; Kim, D. A Depth Video-based Human Detection and Activity Recognition using Multi-features and Embedded Hidden Markov Models for Health Care Monitoring Systems. *Int. J. Interact. Multimed. Artif. Intell.* **2017**, *4*, 54–62. [CrossRef]
40. Jalal, A.; Kamal, S.; Kim, D.-S. Detecting complex 3D human motions with body model low-rank representation for real-time smart activity monitoring system. *KSII Trans. Internet Inf. Syst.* **2018**, *12*, 1189–1204.
41. Jalal, A.; Kamal, S. Improved Behavior Monitoring and Classification Using Cues Parameters Extraction from Camera Array Images. *Int. J. Interact. Multimed. Artif. Intell.* **2019**, *5*, 71. [CrossRef]
42. Jalal, A.; Quaid, M.A.K.; Hasan, A.S. Wearable sensor-based human behavior understanding and recognition in daily life for smart environments. In Proceedings of the 2018 International Conference on Frontiers of Information Technology (FIT), Islamabad, Pakistan, 17–19 December 2018; pp. 105–110.
43. Mahmood, M.; Jalal, A.; Sidduqi, M.A. Robust spatio-Temporal features for human interaction recognition via artificial neural network. In Proceedings of the 2018 International Conference on Frontiers of Information Technology (FIT 2018), Islamabad, Pakistan, 17–19 December 2018.
44. Jalal, A.; Quaid, M.A.K.; Sidduqi, M.A. A Triaxial acceleration-based human motion detection for ambient smart home system. In Proceedings of the 2019 16th International Bhurban Conference on Applied Sciences and Technology (IBCAST), Islamabad, Pakistan, 8–12 January 2019; pp. 353–358.
45. Jalal, A.; Mahmood, M.; Hasan, A.S. Multi-features descriptors for human activity tracking and recognition in Indoor-outdoor environments. In Proceedings of the 2019 16th International Bhurban Conference on Applied Sciences and Technology (IBCAST), Islamabad, Pakistan, 8–12 January 2019; pp. 371–376.
46. Jalal, A.; Mahmood, M. Students' behavior mining in e-learning environment using cognitive processes with information technologies. *Educ. Inf. Technol.* **2019**, *24*, 2797–2821. [CrossRef]
47. Jalal, A.; Nadeem, A.; Bobasu, S. Human Body Parts Estimation and Detection for Physical Sports Movements. In Proceedings of the 2019 2nd International Conference on Communication, Computing and Digital Systems (C-CODE 2019), Islamabad, Pakistan, 6–7 March 2019.
48. Jalal, A.; Quaid, M.A.K.; Kim, K. A wrist worn acceleration based human motion analysis and classification for ambient smart home system. *J. Electr. Eng. Technol.* **2019**, *14*, 1733–1739. [CrossRef]
49. Ahmed, A.; Jalal, A.; Kim, K. Region and decision tree-based segmentations for multi-objects detection and classification in outdoor scenes. In Proceedings of the 2019 International Conference on Frontiers of Information Technology (FIT 2019), Islamabad, Pakistan, 16–18 December 2019.
50. Rafique, A.A.; Jalal, A.; Kim, K. Statistical multi-objects segmentation for indoor/outdoor scene detection and classification via depth images. In Proceedings of the 2020 17th International Bhurban Conference on Applied Sciences and Technology (IBCAST), Islamabad, Pakistan, 14–18 January 2020; pp. 271–276.
51. Ahmed, A.; Jalal, A.; Kim, K. RGB-D images for object segmentation, localization and recognition in indoor scenes using feature descriptor and Hough voting. In Proceedings of the 2020 17th International Bhurban Conference on Applied Sciences and Technology (IBCAST), Islamabad, Pakistan, 14–18 January 2020; pp. 290–295.
52. Tamara, S.; Akhter, I.; Suliman, A.; Ghadi, Y.; Jalal, A.; Park, J. Pedestrian Physical Education Training over Visualization Tool. *CMC* **2022**, *73*, 2389–2405.
53. Quaid, M.A.K.; Jalal, A. Wearable sensors based human behavioral pattern recognition using statistical features and reweighted genetic algorithm. *Multimed. Tools Appl.* **2020**, *79*, 6061–6083. [CrossRef]
54. Nadeem, A.; Jalal, A.; Kim, K. Human Actions Tracking and Recognition Based on Body Parts Detection via Artificial Neural Network. In Proceedings of the 3rd International Conference on Advancements in Computational Sciences (ICACS 2020), Lahore, Pakistan, 17–19 February 2020.

55. Badar Ud Din Tahir, S.; Jalal, A.; Batool, M. Wearable Sensors for Activity Analysis using SMO-based Random Forest over Smart home and Sports Datasets. In Proceedings of the 3rd International Conference on Advancements in Computational Sciences (ICACS 2020), Lahore, Pakistan, 17–19 February 2020.
56. Rizwan, S.A.; Jalal, A.; Kim, K. An Accurate Facial Expression Detector using Multi-Landmarks Selection and Local Transform Features. In Proceedings of the 2020 3rd International Conference on Advancements in Computational Sciences (ICACS), Lahore, Pakistan, 17–19 February 2020; pp. 1–6.
57. Ud din Tahir, S.B.; Jalal, A.; Kim, K. Wearable inertial sensors for daily activity analysis based on adam optimization and the maximum entropy Markov model. *Entropy* **2020**, *22*, 579. [CrossRef] [PubMed]
58. Alam; Abduallah, S.; Akhter, I.; Suliman, A.; Ghadi, Y.; Tamara, S.; Jalal, A. Object detection learning for intelligent self automated vehicles. In Proceedings of the 2019 IEEE International Conference on Vehicular Electronics and Safety (ICVES), Cairo, Egypt, 4–6 September 2019.
59. Jalal, A.; Khalid, N.; Kim, K. Automatic recognition of human interaction via hybrid descriptors and maximum entropy markov model using depth sensors. *Entropy* **2020**, *22*, 817. [CrossRef]
60. Batool, M.; Jalal, A.; Kim, K. Telemonitoring of Daily Activity Using Accelerometer and Gyroscope in Smart Home Environments. *J. Electr. Eng. Technol.* **2020**, *15*, 2801–2809. [CrossRef]
61. Jalal, A.; Batool, M.; Kim, K. Stochastic Recognition of Physical Activity and Healthcare Using Tri-Axial Inertial Wearable Sensors. *Appl. Sci.* **2020**, *10*, 7122. [CrossRef]
62. Jalal, A.; Quaid, M.A.K.; Kim, K.; Tahir, S.B.U.D. A Study of Accelerometer and Gyroscope Measurements in Physical Life-Log Activities Detection Systems. *Sensors* **2020**, *20*, 6670. [CrossRef]
63. Rafique, A.A.; Jalal, A.; Kim, K. Automated Sustainable Multi-Object Segmentation and Recognition via Modified Sampling Consensus and Kernel Sliding Perceptron. *Symmetry* **2020**, *12*, 1928. [CrossRef]
64. Ansar, H.; Jalal, A.; Gochoo, M.; Kim, K. Hand Gesture Recognition Based on Auto-Landmark Localization and Reweighted Genetic Algorithm for Healthcare Muscle Activities. *Sustainability* **2021**, *13*, 2961. [CrossRef]
65. Nadeem, A.; Jalal, A.; Kim, K. Automatic human posture estimation for sport activity recognition with robust body parts detection and entropy markov model. *Multimed. Tools Appl.* **2021**, *80*, 21465–21498. [CrossRef]
66. Akhter, I. Automated Posture Analysis of Gait Event Detection via a Hierarchical Optimization Algorithm and Pseudo 2D Stick-Model. Master's Thesis, Air University, Islamabad, Pakistan, December 2020.
67. Ud din Tahir, S.B. A Triaxial Inertial Devices for Stochastic Life-Log Monitoring via Augmented-Signal and a Hierarchical Recognizer. Master's Thesis, Air University, Islamabad, Pakistan, December 2020.
68. Benitez-Garcia, G.; Olivares-Mercado, J.; Sanchez-Perez, G.; Yanai, K. IPN hand: A video dataset and benchmark for real-time continuous hand gesture recognition. In Proceedings of the 2020 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, 10–15 January 2021; pp. 4340–4347.
69. Materzynska, J.; Berger, G.; Bax, I.; Memisevic, R. The jester dataset: A large-scale video dataset of human gestures. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, Seoul, Korea, 27–28 October 2019.
70. Yamaguchi, O.; Fukui, K. Image-set based Classification using Multiple Pseudo-whitened Mutual Subspace Method. In Proceedings of the 11th International Conference on Pattern Recognition Applications and Methods, Vienna, Austria, 3–5 February 2022.
71. Zhou, B.; Andonian, A.; Oliva, A.; Torralba, A. Temporal relational reasoning in videos. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 803–818.
72. Gammulle, H.; Denman, S.; Sridharan, S.; Fookes, C. TMMF: Temporal Multi-Modal Fusion for Single-Stage Continuous Gesture Recognition. *IEEE Trans. Image Process.* **2021**, *30*, 7689–7701. [CrossRef] [PubMed]
73. Shi, L.; Zhang, Y.; Hu, J.; Cheng, J.; Lu, H. Gesture recognition using spatiotemporal deformable convolutional representation. In Proceedings of the 2019 IEEE International Conference on Image Processing (ICIP), Taipei, Taiwan, 22–25 September 2019; pp. 1900–1904.
74. Kopuklu, O.; Kose, N.; Rigoll, G. Motion fused frames: Data level fusion strategy for hand gesture recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Salt Lake City, UT, USA, 18–22 June 2018; pp. 2103–2111.
75. Benitez-Garcia, G.; Prudente-Tixteco, L.; Castro-Madrid, L.C.; Toscano-Medina, R.; Olivares-Mercado, J.; Sanchez-Perez, G.; Villalba, L.J.G. Improving real-time hand gesture recognition with semantic segmentation. *Sensors* **2021**, *21*, 356. [CrossRef] [PubMed]
76. Jalal, A.; Batool, M.; Kim, K. Sustainable wearable system: Human behavior modeling for life-logging activities using K-Ary tree hashing classifier. *Sustainability* **2020**, *12*, 10324. [CrossRef]