Received December 3, 2021, accepted January 13, 2022, date of publication January 18, 2022, date of current version January 27, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3144456

Angular Features-Based Human Action Recognition System for a Real Application With Subtle Unit Actions

JAEYEONG RYU, ASHOK KUMAR PATIL, BHARATESH CHAKRAVARTHI, ADITHYA BALASUBRAMANYAM, SOUNGSILL PARK, AND YOUNGHO CHAI
Graduate School of Advanced Imaging Science, Chung-Ang University, Seoul 06974, South Korea

Corresponding author: Youngho Chai (yhchai@cau.ac.kr)

This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (Ministry of Science and ICT) (No. 2018-0-00599, Proxemics based Pervasive Interactions for the Wide area·High speed·Serial Motion Recognition); and in part by the Chung-Ang University Research Scholarship Grants, in 2021.

ABSTRACT Human action recognition (HAR) technology is receiving considerable attention in the field of human-computer interaction. We present a HAR system that works stably in real-world applications. In real-world applications, the HAR system needs to identify detailed actions for specific purposes, and the action data includes many variations. Accordingly, we conducted three experiments. First, we tested our recognition system's performance on the UTD-MHAD dataset. We compared our system's accuracy with results from previous research and confirmed that our system achieves a 91% average performance among recognition systems. Furthermore, we hypothesized the use of a HAR system to detect burglary. In the second experiment, we compared the existing benchmark data with our crime detection dataset. We recognized the test scenarios' data by using the recognition system trained by each dataset. The recognition system trained by our dataset achieved higher accuracy than the past benchmark dataset. The results show that the training data should contain detailed actions for a real application. In the third experiment, we tried to find the motion data type that stably recognizes action regardless of data variation. In a real application, the action data are changed by people. Thus, we introduced variations in the action data using the cross-subject protocol and moving area setting. We trained the recognition system using each position and angle data. In addition, we compared the accuracy of each system. We found that the angle format results in better accuracy because the angle data are beneficial for converting the action variation into a consistent pattern.

INDEX TERMS Human action recognition, skeleton, motion capture, ELM classifier, surveillance.

I. INTRODUCTION

There have been many studies on human action recognition (HAR), which is expected to play a major role in human–computer interaction [1]. HAR research is applicable to many fields where computers and people interact. For instance, HAR is used in surveillance systems [2]. Such systems can identify people's actions captured by sensors. Biomedical diagnosis also uses recognition technology. In [3], patients' gaits were analyzed using HAR technology. Although existing HAR systems mainly use machine learning algorithms, in recent times, many researchers have attempted to use deep learning-based algorithms [4].

HAR technology requires human action data. In many HAR studies, RGB data were captured by using vision-based sensors [5]. Cameras was usually utilized as the vision method, but depth information was missing from the data. This concern was addressed by using Kinect sensors and stereo cameras, which are capable of capturing depth data [6]. Other researchers used Motion Capture systems. In such systems, markers are attached to the user and the marker's position is continuously collected [7]. Inertial Measurement Unit (IMU) sensors, which can extract a human's orientation data precisely, can also be used in sensing systems [8]. The human action-sensing system of HAR technology has developed such that it can extract data more accurately.

The sensed data are then processed by the HAR system. Many proposed HAR systems generate their own unique

The associate editor coordinating the review of this manuscript and approving it for publication was Sambit Bakshi.

feature vectors from the raw data. The forms of raw data are commonly classified into two categories. The first is the skeleton data, containing the position of all joints. In vision sensor cases, RGB-D data is used for tracking each joint position [9]. The second is the angle data format, composed of the rotation of each body segment. Angle data can be extracted from skeleton data by using the vector dot product [10]. These data are also obtained by using IMU sensors [8]. Open benchmark datasets usually provide action data as skeleton data or as recorded videos [11].

In this study, we used the angle data of the Motion-Sphere as the recognition system's input data. Motion-Sphere is a technique for visualizing motion [12]. A person's movements are marked as a trajectory on the sphere. The uploaded trajectory can be edited, and it is possible to author new motions using Motion-Sphere. The trajectory can be interpreted at the θ and ϕ angles based on the spherical coordinate system. We used those angle data as input data for our recognition system. The inputted angle data are transformed into feature vectors that contain spatial and temporal information. The final classifiers use those feature vectors. We utilized an Extreme Learning Machine (ELM) as the classifier. ELMs have been applied to HAR systems [13]. They require much lower training time than other ML algorithms because they use only one hidden layer. Thus, we created several ELM classifiers and trained them using the same training data. Our system gathered the predicted results of all ELMs and recognized the inputted action data as a class label.

We tried to find ways to create a stable recognition system in real-world application situations. We mainly focused on the training data and action data format. In applying the HAR, we believed that the HAR system required subtle motions that meet specific purposes in the training data. To test our hypothesis, we created test scenarios' data to allow surveillance systems to detect burglaries. We trained the recognition system using the existing benchmark dataset and recognized the test data. However, the recognition system failed to detect crime scenarios because the existing benchmark was designed only to evaluate the recognition system. Therefore, we created a new dataset containing specific subtle unit actions for detecting the burglary. In the experiment section, we compare the prediction results of the systems trained using each training data. After the training data experiment, we analyze the effect of data type in real applications. We assume that variations in the motion data will be introduced by a person. For example, many variations will occur for the same movement in the HAR systems used by various people. We designed our experiments to test the above situation virtually. In the experiment, we excluded the effects of sensing environments. Therefore, to reduce the impact on the environment, we captured existing benchmark data using our sensing environment. In addition, we used an average level of the recognition system, which can represent existing recognition systems. Through the above setting, we could study the effect of data format on recognition in various people's data. Finally, we seek the development direction of training data and motion data type for real applications with the experiment results.

After this introduction section, we briefly introduce the overall HAR studies in Section 2, such as sensing environment, benchmark data, and recognition systems. In Section 3, we explain our recognition system, test scenarios, and training data. Section 4 consists of three experiments. The first is to evaluate the recognition system performance. After that, we compare the existing benchmark data set with the data set for detecting burglary crime in test scenarios. The third experiment analyzes the optimal data formats between angular data and position data in real-world applications. In the final section, we summarize the main contributions of the paper and our future study.

II. RELATED WORK

A. SENSING SYSTEM AND BENCHMARK DATA SET

Early motion recognition researchers acquired human motion data using video cameras. They tried to extract significant features in the recorded video [14]. As the price of stereo cameras decreased, researchers began to use depth information as well. The depth data provided new features [15]. In particular, skeleton data has increased substantially since the launch of Kinect cameras [16]. There are other ways to collect action data without a camera. In the motion capture (Mocap) system, the optical markers were attached to the joints of a user [17]. The locations of the user's joints are collected continuously. This system's advantages are obtaining precise position data and few noise data. However, the Mocap system was not widely distributed because of the high cost of equipment consumption and complex configuration. Another sensing system uses inertial measurement unit (IMU) sensors. This sensor accepts human movement as orientation information [18]. The IMU sensor has the advantage of sensing accurate data, like motion capture equipment. At present, researchers are attempting to design complex sensing environments that use multimodality, because a complex system can collect more accurate motion data [19]. We gather human motion data using a multimodal sensing system [20]. The system was composed of one Lidar sensor and ten IMU sensors.

Some researchers shared the motion data used in their recognition systems [21]. HAR researchers adopted a specific dataset that was commonly used to evaluate their system recognition performance compared with other systems. The widely used dataset became benchmark data [22]. This study mainly focuses on the benchmark dataset that provides skeleton data rather than the benchmark data that uses images.

Some widely used benchmark data are the MSR aciton3D [21], MSR-DailyActivity3D [22], UTD-MHAD [23], and UTKinect [24]. They commonly used the Kinect camera to extract skeleton data. The benchmark datasets generally comprise movements that occur during daily life actions. Thus, some actions of the benchmark overlap with other benchmark data, such as walking, jogging, squatting, pushing, and pulling. The NTU benchmark was published recently [25]. This dataset has a large amount of motion data with 120 classes, most of which relate to daily actions. In contrast, a few benchmarks were designed for specific purposes [26]- [27]. In one study, the benchmark data consisted of 15 exercise actions aimed at patient self-management of cardiovascular disease [26]. The other case is the Microsoft Research Cambridge-12 (MSRC-12) [27]. This dataset was composed of 12 actions for controlling Kinect's game. We developed our dataset to help a surveillance system detect burglaries. It consists of 11 actions and has detailed motions related to the detection of burglary. We captured the dataset using our sensing system. Our motion data is accurate because it is made of quaternion data. After analyzing the motion data accurately, we will use a camera-based sensing system for real-world applications. The camera-based sensing system is used for practical usage because of its convenience.

B. HUMAN ACTION RECOGNITION SYSTEM

Human action recognition research can be classified as per data type. Based on skeleton data, we divided the studies into two categories. The first involves a recognition system that uses the position data stored in the skeleton. In that method, data preprocessing, wherein the size of the skeleton model and the initial rotation of the skeleton data are normalized, is essential [28]. Previous studies extracted feature vectors from location data after data preprocessing. Usually, they attempted to extract temporal and spatial characteristics from position data. A representative time feature is the velocity value, which can be obtained by position variation of the previous frame and the current frame [29]. Some studies adopted relative position data. They calculated spatial differences between joints on a frame [30]. A HAR study used a trajectory that included temporal and spatial characteristics simultaneously [31]. Other researchers attempted to change the data's manifold [32]. They transferred the position data to another space to recognize benchmark actions. In this case, the unique feature vector was extracted from the skeleton data.

The second method is to utilize angle data. The skeleton data's joint points are subtracted from each other, and the point's location data transformed into vectors. Angle data are extracted by calculating the angle between each body vector. The angle data are directly obtained when we capture the subject's action using IMU sensors. In this method, normalization is not required. In [33], the 3D position data was transformed to spherical coordinates. After transformation, the angle data was used to extract joint angles similarities (JAS) features. The sequence of the most informative joints (SMIJ) features are obtained between the angle of the two joint vectors in [34]. They gathered all the angles information and selected the most valuable angle to identify the action. In [35], angle data was used on the spherical coordinate. Angle data was converted to modified spherical harmonics (MSH) values. In [10], the angle of the joint vector was also calculated. Only six angles were extracted from the skeleton data. Angle data type is less affected by the physical characteristics of users [34].

In many studies, location data and angle data are used as input data. These studies examined how to extract significant features from each data type. We can freely change the position data to angle data if we have skeleton data. There are advantages of each data format. In [36], both 3D normal position and joint angle data were used. We experimented with both data formats to find the optimal data type for actual applications. In the experiment, we compared the recognition results obtained by each data type. We analyzed the effect of each data format by using the experimental results. Finally, we derived the proper data format that stably identifies the inputted action in real-world applications.

III. PROPOSED SYSTEM

In this section, we describe our action recognition system and action data. Our recognition system can extract position and angle data types from inputted raw data. We will compare the accuracy of each data type via experiments. The action data are the training data and test scenario data. We designed the test scenarios to test our recognition system in actual applications. We created two training datasets to classify the test scenario's action. Thus, we can check the effect of the training data in the test scenario.

A. ACTION RECOGNITION PROCESS

Our recognition consists of three main stages. The first stage is data preprocessing. The inputted raw data formats differ depending on the experiment. The system converts the raw data into the normalized unit motion vector (NUM vector) in this stage. After getting NUM vectors, the system extracts the Motion-Sphere's angle from the NUM vector during feature extraction. The NUM vector is used as a feature of the position in the recognition system. The NUM vector is the position data type, and the Motion-Sphere's angle is the angle data type. The system concatenates the extracted features in a feature vector form. The final stage is the classification part. We adopted extreme learning machine (ELM) as our classifier. The ELM can train and test the data quickly, as it has only a single layer. Therefore, we constructed an ensemble method of three ELM classifiers. The final classification result comes from three ELM's prediction values.

1) DATA PREPROCESSING

Data preprocessing methods differ depending on the inputted raw data formats. We adopted two data formats as per the experiment. The first experiment used skeleton data from the UTD-MHAD benchmark dataset. The other experiments used our training data and test scenario data obtained by our sensing system. This sensing system acquired the user's body segment rotation data by using IMU sensors. The sensors obtain the motion data as quaternion data.

In the case of quaternion data, we used the rotation of the quaternion. We applied the quaternion's rotation to the initial unit vector. The unit vector is the same as the user's initial

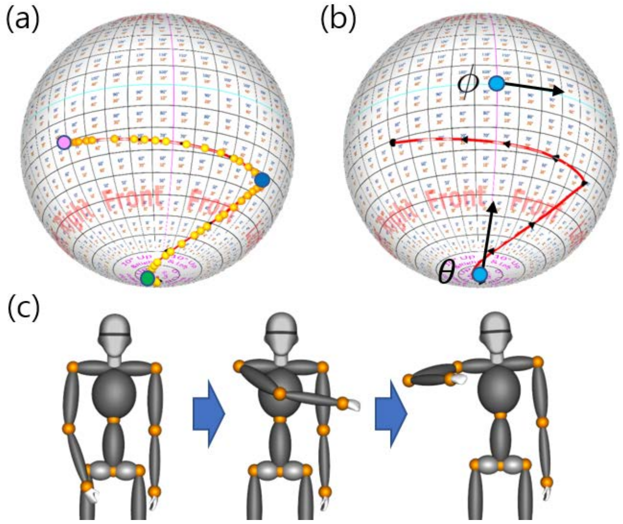

FIGURE 1. Motion-Sphere and avatar's action example, (a) Motion-sphere frame points: the Crowbar-right action (green point:16frame, blue point:38frame, pink point:62frame) (b) Motion-sphere's trajectory (right upper arm), the directions and the origin of θ and ϕ angles (c) Avatar's action: Crowbar-right, the first avatar shows 16frame, the next avatar shows 38frame, and the final avatar shows 62frame.

pose. Then, the vector's rotation is the same as the user's actual action. We obtained the NUM vector by rotating the unit vector. We captured the user's motion using 10 IMU sensors. Therefore, 10 NUM vectors are extracted in this process. The skeleton data is simpler than the previous quaternion data. We calculate a joint vector by subtracting two points of the upper and lower limb. After that, we apply the vector's normalization to the joint vector. We selected 20 joint points from inputted skeleton model. We obtained 10 NUM vectors from the joint points. Finally, the number of result vectors is the same in the quaternion and the skeleton.

2) FEATURE EXTRACTION AND RECOGNITION

In the feature extraction, the system extracts angle features from the NUM vector. The extracted angle features are the Motion-Sphere's angles. As mentioned earlier, Motion-Sphere is a technique that visualizes human movements. We can draw Motion-Spheres based on the number of joints in the action data. In our case, we use 10 IMU sensors, so we express the user's behavior in 10 Motion-Spheres. Figure 1 visualizes the Crowbar-right motion. Their visualization methods are different. The first row shows the Motion-Sphere of the right upper arm; the second row is the action by virtual avatar in Figure 1 (c).

The Motion-Sphere expresses the avatar's action as frame points (a) and trajectory (b). The yellow point represents each frame point, and the red line denotes the joint's moving path in Figure 1 (b). The avatar's movement is 16frame, 38frame, and 62frame in Figure 1 (c). We can see the avatar's movement as green point (16frame), blue point (38frame), and pink point (62frame). Thus, an observer can understand the action data's movement by using the Motion-Sphere.

The Motion-Sphere's angle data consist of θ and ϕ angle data. The θ and ϕ angles are shown in Figure 1 (b). The θ angle represents vertical movements. We set the bottom of the sphere as θ angle's 0°, as in Figure 1 (b). The ϕ angle expresses horizontal movement in Figure 1 (b). The phi's origin point is the user's front side at the standing pose. We obtain two angle values by the NUM vector. We transform the frame point to θ and ϕ angle values by using the vector dot product. The angle between a starting NUM vector and a current NUM vector is the current angle value. Through these methods, we convert the 10 NUM vectors into the Motion-Sphere's angles.

The final feature vector forms differ according to the data type. The angle feature vector consists of the Motion-Sphere's angle and relative angle. The relative angle is calculated by two NUM vectors. The angle expresses the spatial characteristics of the action. The position feature vector is composed of the NUM vectors. The feature already includes spatial characteristics. The additional spatial feature is unnecessary in position data type. We will compare these two feature vectors in the third experiment. Through the comparison, we will find a suitable data type in real-world application.

We classified the feature vectors using an ELM algorithm. Our system's feature vector is simple and contains essential information identifying inputted action. In these kinds of features, a complex recognition system is excessive. The ELM algorithm's configuration is light, but it can clearly distinguish the differences between our feature vectors. The ELM algorithm can train the feature vectors rapidly. Thus, we used three ELM structures. All the ELM training conditions are the same. The final class label is decided by the average of the three ELM prediction values. The recognition test results show that the accuracy of the structure using multiple ELMs was higher than that of a single ELM.

B. NEW DATA SET AND TEST SCENARIOS

We designed our new test scenarios to test the recognition system in real-world application. The scenarios consist of three detailed unit actions for detecting burglarious actions. Table 1 summarizes our test scenario's configurations. We set two-door shapes depending on the opening direction. The first column shows the door's shape. The scenario's actions are changed depending on the door's shape. In the outward door, the door opening action is the pull action, and the crime action is the Crowbar-right. In the other door, the actions' directions are the opposite (the Pull action is changed to the Push action, and Crowbar-right is switched to Crowbar-left) Crowbars are still widely used in crimes [37]. Hence, we used crowbar actions in crucial movements to determine the occurrence of a crime. Both crowbar actions are shown in Figure 2.

We paired the crime and non-crime scenarios. All paired sets are composed of similar motions. In Table 1, the first and second rows are paired, and the remaining rows are also paired. At the first scenario pair, the Pull door lock action is similar to the Crowbar-right action. The difference between

TABLE 1. Test scenarios including unit actions for the surveillance system.

Opening direction	Crime	Unit Action 1	Unit Action 2	Unit Action 3
Outward	Non-crime	Password	Pull door lock	Pull and enter
	Crime	Knock	Crowbar-right	Pull and enter
Inward	Non-crime	Key	Lever door lock	Hand push and enter
	Crime	Knock	Crowbar-left	Body push and enter

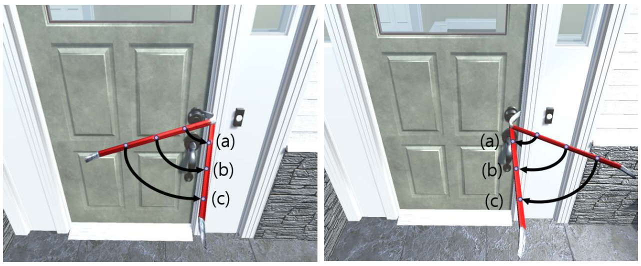

FIGURE 2. Crowbar action variation by various subjects, A: Crowbar-right, B: Crowbar-left, (a): Subject1, (b): Subject2, (c): Subject3.

the two motions is the direction of pulling. We show both pulling actions in the virtual avatar's action in Figure 3 (a) and (c). The Pull door lock action is pulled from the door lock to the user, and the Crowbar-right action is the movement from the left side to the right side. The recognition system should distinguish two actions to identify the crime. In the case of the other pair, the crucial actions are the Lever door lock and Crowbar-left. Both actions' starting points are similar, but the pulling directions are different. After the starting point, the user's hand goes down at the Lever door lock. When the user takes the Crowbar-left action, the user's hand moving direction is toward the body from the handle.

After creating the test data, we recognized the scenario data using our recognition system. To train the recognition system, we made an existing benchmark reference (EBR) dataset. The EBR dataset is composed of the actions extracted by three existing benchmark datasets. We selected the action to recognize the test scenario data among the UTD-MAHD, MSR aciton3D, and UTKinect datasets. The selected action and used benchmark dataset are presented in Table 2. We trained the recognition system using the EBR dataset and used the system to classify the scenarios' data. The trained system misclassified the subtle unit action. The classification results are covered in detail in the experiment section.

We assumed that the misclassification comes from missing information of the detailed action. Thus, we created a new dataset for detecting burglary crime (DBC) in test scenarios. The newly generated DBC dataset can be found in the last column of Table 2. At first, we retained the motions that were useful to detect the test scenario's action among the EBR dataset. We added more detailed actions to detect burglary cases. We included new crowbar actions in Table 2. The new dataset consists of 11 actions. We compare the datasets in Table 2. For the Pull action, there are four detailed actions in the DBC dataset. We placed all the Pull actions in the same row, and we classified subtle pull actions as a column in the DBC dataset. All pull action variations are shown in Figure 3. Similarly, there is one push action in the EBR dataset. The detailed actions are 'Body push and enter' and 'Hand push and enter'. We also placed the same category action in the same row and classified them into columns. Those actions are shown in Figure 4. In Table 2, the Key and Lever door lock actions occupy two rows in the DBC dataset column. The EBR movements belonging to the two rows of the DBC movement are similar to the DBC action. The Key action's shape is similar to the Draw circle actions' shape. The overall shape looks like drawing a circle, but their radii are different.

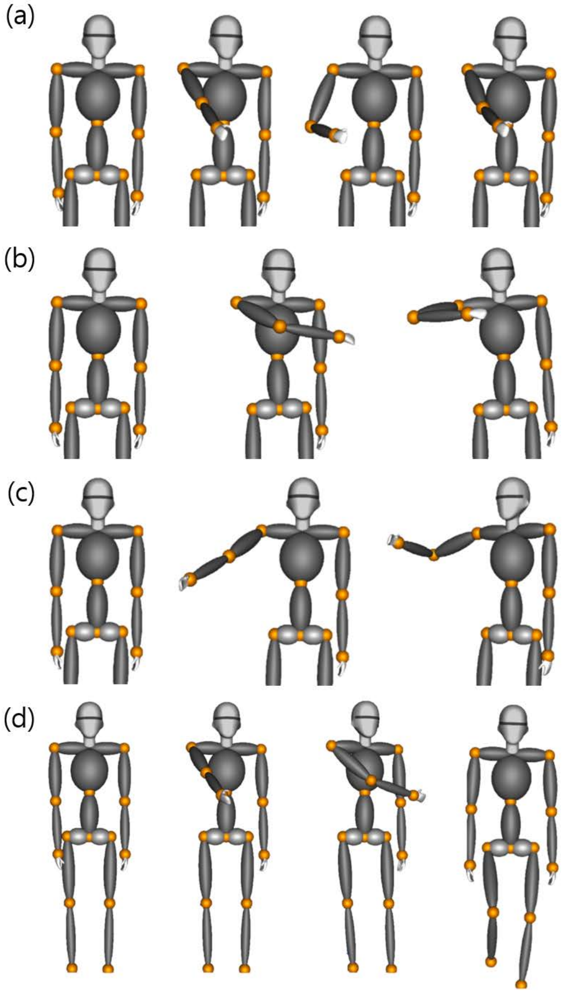

FIGURE 3. Pull actions of the DBC data set, (a) Pull door lock, (b) Crowbar-right, (c) Crowbar-left, (d) Pull and enter.

TABLE 2. Rreference dataset combining existing benchmark data (EBR data) and the new dataset for detecting burglary crime with detailed actions (DBC data).

No.	Benchmark	EBR data set	DBC data set			
1	MSR-Action	Horizontal arm wave	1. Password			
2	UTKinect	Pull	2. Pull door lock	3. Pull and enter	4. Crowbar-right	5.Crowbar-left
3	UTD-MHAD	Push	6. Body push and enter		7. Hand push and enter	
4		Draw circle	8. Key			
5		Draw circle counter				
6		Swipe left	9. Lever door lock			
7		Swipe right				
8		Walk	10. Walk			
9		Knock	11. Knock			
10		Arm cross				
11		Arm curl				

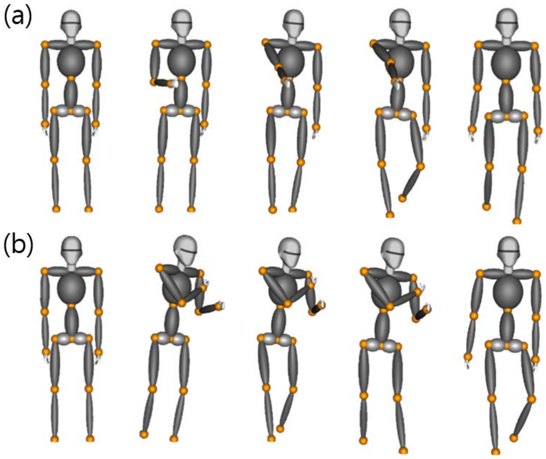

FIGURE 4. Push actions of the DBC data set, (a) Hand push and enter, (b) Body push and enter.

In the Lever door lock action, their moving trajectories are similar. We changed the EBR's general action to subtle action for a specific application. We also tested the recognition system trained by the DBC set. The system could detect the burglary case accurately in the test scenarios. We compare the recognition accuracy of each the DBC and the EBR dataset in the experiment section.

IV. EXPERIMENT RESULT

We simulated the application of the HAR system to analyze our main concerns. We mainly focused on two aspects. The first is the training data of the recognition system. In a supervised learning system, the system's performance varies depending on the training data. We studied the requirements of training data for practical applications. The second is the action data type of the recognition system inside. We can express the movement of a human using position and angle data types. We can extract features from the position and angle data types within the recognition system. We analyzed the advantages and disadvantages of each data type in our test scenarios where the HAR system is used to recognize subtle actions. We designed three experiments to analyze the aforementioned two aspects.

Before examining the main points, we objectively evaluated the level of our recognition system through the first experiment, which used existing UTD-MHAD data. We compared our system's accuracy with other studies. Based on the results, we can check whether our system can achieve the average performance of the HAR system.

The first point, the training data, was experimented with in the second experiment, in which the recognition system trained using existing benchmark data was utilized for the real-world application. As described in Section 3, we created a scenario test data that suits the practical application. We adopted the test data to check the trained recognition system. The existing benchmark was used to evaluate the performance of the recognition systems; therefore, it lacks the specific actions required in each application area. We categorized the cases that may occur in this situation, and we summarized the recognition results of the recognition system.

The third experiment was conducted to test the data type of the recognition system. We compared the accuracy of position as well as angle data. In the surveillance system, the system identifies the action of untrained people. Furthermore, people can perform the same action using slightly different movements. Even in this case, the recognition system should operate stably. We attempted to analyze the accuracy based on various data formats in a complex situation.

A. UTD-MHAD DATASET

We tested our recognition system using the University of Texas at Dallas Multimodal Human Action Dataset (UTD-MHAD) data. UTD-MHAD data comprises 27 motions. We used a cross-subject protocol in this experiment [23]. Thus, odd numbers of subjects data were used as training data, while even numbers were used as test data. We extracted our features from the skeleton data of UTD-MHAD. The features were the Motion-Sphere's angle data, the temporal variation of the angle, and the relative angle. We obtained the Motion-Sphere's angle features mainly from the right arm because the UTD-MHAD data have many right-arm actions. We extracted spatial features by using relative angles, consisting of the pelvis, right leg, and left arm movements. We searched the number of the hidden layer by an experi-

TABLE 3. Accuracy comparison with previous research on UTD-MHAD.

Method	Accuracy (%)
JTM (Joint Trajectory Maps) [38]	85.81
JDM (Joint Distance Maps) [39]	88.10
FV-ELM (Fisher Vector-ELM) [40]	92.6
HDM (Hierarchical Dynamic Model) [41]	92.80
SEMN (Skeleton Edge Motion Networks) [42]	95.57
Our method	91.67

mental method. We could obtain the highest accuracy when we used 4,500 layers in our algorithm.

We wanted to conduct experiments using a recognition system representative of existing systems. The existing recognition systems can recognize the inputted action with high or low accuracy. We compare the accuracy of our system and other systems in Table 3. Our system achieved a moderate level of performance in the comparison. After confirming the comparison result, we experimented on the training data and data type.

As the recognition performance of recognition systems is improving steadily, we will use a recognition system with higher accuracy in the future study. The SEMN method achieved the best accuracy in our comparison, as presented in Table 3. SEMN used the skeleton motion networks based on the CNN. The JDM method also adopted CNN but achieved less accurate results than our method. SEMN was published more recently than JDM. We verified that, at first, deep learning achieved lower performance than machine learning. However, these days, deep learning-based approaches achieve higher accuracy than machine learning-based ones. Thus, we plan to utilize a deep learning-based algorithm in future research.

We learned the features used in each system while investigating the accuracy of the existing studies. The features between various studies are different. Even with the same recognition system, the used features differ depending on the benchmark dataset. In deep learning cases, hyperparameters were changed by the dataset. However, actions are changed by people and the environment in a real application. Thus, we should steadily adjust the current HAR system's features and hyperparameters to cope with action variation. To do so, we need to collect enough training data to handle all possible action variations. The data collection method is difficult to implement realistically. In this situation, we hypothesize that if we have a definition of the action, we can cope with all possible variations. The definition should consistently keep the action's features even if the human performer and environment are changed. We tried to find the definition of the action in this work. In the third experiment, we examined which data type is optimal for defining actions.

B. TRAINING DATA COMPARISON RESULTS

In this experiment, our main purpose is to study the training dataset for a real-world application. As mentioned earlier, we assumed that the benchmark data is unsuitable as training data in the application system, because the current benchmark was intended for the recognition system's performance evaluation. We confirmed our hypothesis through an experiment. As mentioned earier, we developed a reference benchmark (i.e., EBR dataset) by using a combination of the published benchmark data. The motion selection criteria are the actions identifying our test scenarios. The EBR dataset is composed of 11 actions given in Table 2. The number of actions is the same as the generated DBC dataset. Most of the motions are similar to our DBC dataset, but the DBC dataset contains more detailed actions.

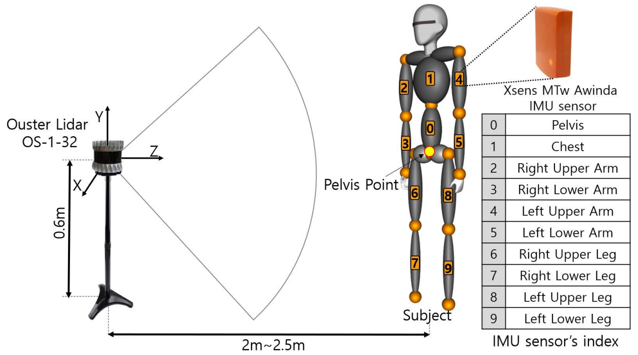

FIGURE 5. Action data sensing system, left: sensing environment, right: IMU sensor and IMU sensor's index.

In the experiment, we directly captured the EBR and DBC datasets using our sensing system. The sensing system's configuration and IMU sensor indexes are shown in Figure 5. The system consisted of 10 IMU sensors and one Light Detection and Ranging (LiDAR) sensor. In the sensing system, the LiDAR sensor continuously tracks the pelvis point position of the user's body. The Lidar sensor is approximately 2 to 2.5 meters away from the subject. In addition, the IMU sensors obtain the motion data of the subject using quaternion data, which were then used as the input data of the recognition system. In EBR data acquisition, we kept the shapes of the existing benchmark data, only changing the data type and sensing environment. Through this procedure, we could set a consistent environment for the training data acquisition. A subject repeated each action eight times to create training data. The same subject captured four test scenarios' data twice. Using this experimental setup, we could test the effect of the training data on the recognition results without human variation.

The accuracy was 100% when we used the recognition system trained upon the DBC dataset. For the DBC dataset, there was no error because the subject and actions in the test data were the same. However, many errors occurred in experiments using the EBR dataset. The accuracy was 50% in the EBR dataset. The wrong cases can be confirmed through Table 4. In Table 4, we compare the predicted results from both datasets.

We can summarize the results of Table 4 into three categories. The first is the test scenario action included in the

TABLE 4. Comparison of the recognition system prediction results on test scenarios according to training data.

Scenario Number	Training data: EBR dataset			Training data: DBC dataset		
	Unit Action 1	Unit Action 2	Unit Action 3	Unit Action 1	Unit Action 2	Unit Action 3
1	Knock	Pull	Draw circle-counter	Password	Pull door lock	Pull and enter
2	Knock	Swipe right	Walk	Knock	Crowbar-right	Pull and enter
3	Knock	Horizontal arm wave	Push	Key	Lever door lock	Hand push and enter
4	Knock	Pull	Walk	Knock	Crowbar-left	Body push and enter
Accuracy	50%			100%		

EBR dataset. For instance, the Knock action is the first action of scenarios 2 and 4 in Table 1. The system classified the action accurately because the EBR dataset had the Knock action. The second is that the EBR dataset contained actions similar to the scenario action. In this case, the trained system could detect the scenario action but failed to distinguish the subtle difference. For example, the second movement of scenario 1 is the Pull door lock. The predicted result is the Pull action. The motion's general meaning is correct. Thus, when calculating the system's accuracy, this kind of answer was classified as the correct answer. The following example is about complex actions consisting of two motions. Action 3 of scenarios 3 and 4 is the Hand push and enter and the Body push and enter, respectively. Both movements are similar, but the pushing methods are different. The recognition system only detected one motion at a time. In scenario 3, the Hand push action was detected as the Push. However, the system distinguished the Body push as the Walk when the hand movements disappeared. Therefore, the recognition system recognized the inputted action without detailed description in the second case. We also classified this answer as the correct answer because the system did not train the combination of the action.

The last case is the test scenario action absent in the EBR dataset. When the test action is vacant in the training data, the recognition system classifies the inputted action as the most similar action. The crowbar actions are absent in the EBR dataset. The second movement of scenario 2 is the Crowbar-right action, but the system recognized it as Swipe right. In the other example, the system predicted the middle action of scenario 4 as the Pull motion, whereas the ground truth is Crowbar-left. Both crowbar action trajectories are similar to each pull and swipe right. However, the meaning of the action is different. The crowbar actions relate to crime, but pull and swipe right actions are daily motions.

Through this experiment, we could assume that existing benchmark data is used in the actual application system. The results of the experiment can be summarized as three cases. The trained system detects the scenario behavior with 100% accuracy when the action exists in the training data. In the second case, the motions exist in the training data, but the action lacks sufficient detail. Although the system can identify the action, it fails to classify the subtle motion class because of a lack of information. In the final case, the scenario action is absent in the training data. In this case, the recognition system tries to find a similar motion to inputted action. Thus, it predicts the test scenario's action as a similar motion in the training data even if the movements' meanings are different. In short, the second and third cases were both caused by a lack of information.

Previously, the benchmark data consisted of daily actions because it was used for testing the system. We considered the HAR system to use in real applications. We tested the recognition system trained using the existing benchmark data. The misclassified cases occurred owing to the absence of a specific action being represented among the training data. Thus, the training data need to include examples of the action for real-world applications. When a HAR system is used in a specific domain, we need to create a new dataset containing detailed, relevant actions.

C. DATA TYPE COMPARISON RESULTS

In this experiment, we wanted to find the optimal data type for defining the characteristics of the action. We compared the accuracy of angle data type and position data type in the burglary detecting case. Through this comparison, we analyzed each data type's accuracy in the specific application. In addition, we discussed a data type that is useful to define the action features with analysis results.

We set the experiment scenarios as a burglary intrusion in the surveillance system. The test scenario is the same as the one used in the previous training data comparison experiment. The system's training data was the DBC dataset. The experiment was conducted with two protocols. The first is the one-person protocol. In this protocol, we obtained the test scenario and training data from a single person. We designed this protocol to confirm our recognition system's performance, which distinguishes the subtle motion of the test scenario. The second is the cross-subject protocol. In this setup, we used the training data from the one-person protocol but obtained the test scenarios' data from other subjects. We captured the test scenario data twice for each person and created a virtual simulation situation where the system recognizes the action of a new person.

Four subjects participated in the third experiment. The participants were men in their 20s and 30s, having similar physiques. Data from one person was used as learning and test scenario data, while data from the other three people was used only as test scenario data. In the test scenario data acquisition, we set the range of each subject's motion differently. Figure 2 illustrates each subject's moving range in the Crowbar actions. The first path (a) inside Figure 2 is

TABLE 5. Test scenarios recognition results depending on data type.

Protocol	Recognition results	
	Angle	Position
One person	100%	100%
Cross-subject	83%	59%

TABLE 6. Number of errors in each data type. Each action has six instances.

Scenario number	Angle data			Position data		
	Act. 1	Act. 2	Act. 3	Act. 1	Act. 2	Act. 3
1	0	5	1	3	3	2
2	0	0	0	2	4	2
3	2	1	1	2	5	0
4	0	0	2	4	0	2

the range of the first subject's action. The subject executed all the actions within a small range if possible. The second path (b) represented a general motion. The moving area of the second subject was similar to the training data. The last person performed all the movements expansively. Through the above setting, we could compare both data types' recognition accuracy when faced with human action variation.

The environment of the sensing system is the same as in the second experiment. Quaternion data obtained by IMU sensors are used as input data of the recognition system. In the system, the inputted data were converted to angle and position data. We applied various feature extraction methods to each data type. The position's feature consists of the NUM vectors, and the angle's features are composed of the Motion-Sphere's angles. We used the ELM classifier for both data types. The ELM classifier's settings vary based on each data format. We searched the adequate hidden layer number using the empirical method.

Table 5 presents the recognition accuracy derived by each data type. In a one-person protocol, both data types obtain 100% accuracy. This result shows that our recognition system can distinguish detailed actions of the test dataset using either data type. In the cross-subject experiments, each data type's recognition accuracy decreases. The accuracy of the angle data declines 17%, while that of position data decreases by 41%. We examined the recognition of each unit motion to analyze the cause of the accuracy difference in each data type.

In Table 6, we summarized the number of misclassifications in the unit actions of each scenario. As can be seen in Table 6, there are more recognition errors in position data type. We can check each scenario recognition rate through Table 6. In scenario 2, the recognition rate of the two data types are clearly different. The accuracy of the angle data is 100%, while that of position data is approximately 55%. The second unit actions in each scenario are the main motions identifying crime and non-crime. For the second unit motions, the angle data's accuracy is 75%, while position data's accuracy is 50%. The angle data achieved high accuracy in the overall as well as the critical unit actions.

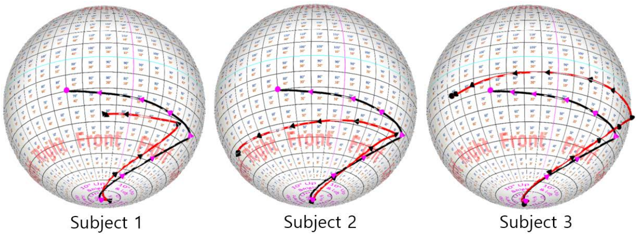

FIGURE 6. Crowbar-right action comparison by the motion-sphere.

We selected the motions that differed substantially in recognition rates in Table 6. The selected actions were the second unit action of scenario 2 and the first unit action of scenario 4. Both unit motions were accurately recognized in angle data, but the accuracy was low in position data. We adopted two data visualization methods. The first is the Motion-Sphere, which visualizes the position data as a trajectory. We compared two trajectories using the Motion-Sphere. The training data are in black color and the subject's data in red color. The second visualization method comprises graphs that show angle and position data. We show the Motion-Sphere's angles and the position value of the cartesian coordinates. We compared the angle's graph with the position's graph.

The second unit action of scenario 2 is Crowbar-right. We express this action using the Motion-Sphere. The Motion-Sphere represents the motion of the right upper arm in Figure 6. We visualize the Crowbar-right action by a virtual skeleton avatar in Figure 3. The training data serves as reference data when we compare the Motion-Spheres in Figure 6. Through the Motion-Sphere, we can confirm that motion pattern varies among subjects. From the left side of the figure, we place the Motion-Spheres of subjects 1 to 3 in order. Thus, the left Motion-Sphere represents the first subject's action. The first subject tried to move as little as possible. The moving area of the trajectory is the smallest. The second subject attempted to act similarly to training data, but the reference and the second subject's trajectories are different. The last subject performed the motion beyond the training data. The Motion-Sphere can express the exact movement of each subject. We can confirm that our experimental setup was successfully applied to action data.

We compared each data graph to find a useful data type for defining the action's pattern. Figure 7 contains four angular graphs and four position graphs, each consisting of training data and subjects. In the upper row, the left is the training data. The right is the first subject's action data. In the lower row, we placed the second and third subject's data in order. First, we analyze the angular graph. The amplitudes of the angular graph are different between subjects, because the subjects' moving areas are different. However, we can observe a consistent pattern in the Crowbar-right action. The Crowbar-right action consists of raising the right arm and pulling left to right. Through the θ angle, we can observe the raising

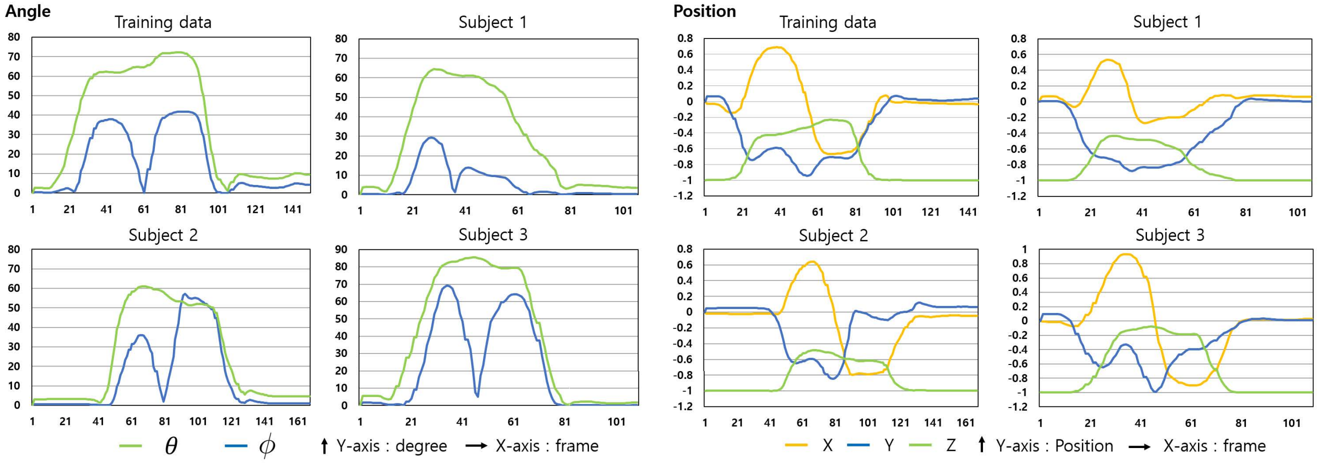

FIGURE 7. Crowbar-right action comparison by angle(left) and position(right) graph.

arm action. The θ angle maintains a certain value and then falls. The arm pulling action passes the origin of ϕ angle when the arm moves from left to right. We took the absolute value of the ϕ angle. Thus, there are two peak points in the case of ϕ angle. Patterns at each angle are consistent in all subjects. After analyzing the angular graph, we examine the position graph, in which it is hard to find consistent action patterns. Specifically, the variations of the y value are different in all subjects' data. Each x and y value looks similar in all experimenters' data. However, the detailed patterns are significantly different because of the action scale variation.

The second example motion is the first unit action of scenario 4. This motion is the Knock action, which consists of three taps on the door. The tapping pattern is the main characteristic of the Knock action. Figure 8 shows each subject's action using the Motion-Sphere. We marked the tapping action area by using a colored region on the trajectory. The pink area represents the tapping action of the training data, and the blue region shows that of the subject's data. Thus, the pink area is fixed, while the blue region varies depending on the subject. All subjects maintained their action characteristic consistently. The first subject's area is smaller than the training data's area. The second subject's area is similar to the training data's region, but there is a small drift difference. The last subject's action is more expansive than the training data's action.

We also display the subject's action data as angular and position graphs in Figure 9. Through the graphs, we can confirm the pattern of the Knock action. In the angle's graph, all graphs have three peak regions in the θ angle. The amplitude of the peak region is proportional to the moving area. Thus, the third subject's data has the largest amplitude in Figure 9. The ϕ angle determines the direction of the Knock action. The training and subject 2 data are slightly different in Figure 8. We can check the drift difference using the ϕ angle graph. The ϕ angle differs by approximately 5° between the training and subject 2. Although the detailed shapes differ between angle graphs, we can derive the common characteristic of the Knock action. From the perspective of θ all subjects' θ graphs have three peak regions. The ϕ angle variation is much less than θ angle variation. The ϕ angle decides the direction of the tapping action. In the position's graph, we can see similar patterns of z values in all subjects. The y value's variations look similar, but their peak values are different in all y graphs. The x value's pattern varies among all x graphs. The ϕ angle and x value patterns differ in all graphs, but the ϕ angle's scale is significantly less than the θ angle's. The system can concentrate on the main tapping action by θ angle's variation. Thus, the recognition system recognized the Knock action accurately in the angle data type.

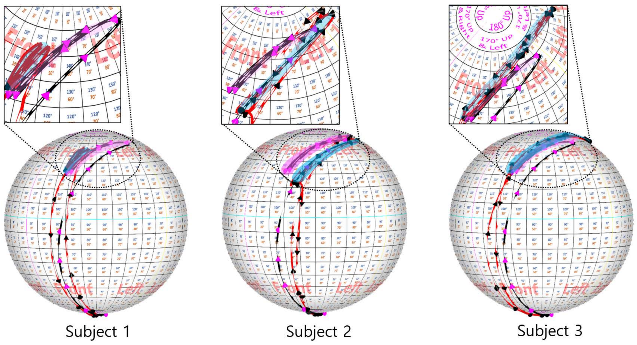

FIGURE 8. Knock action comparison by the motion-sphere.

We analyzed two movements that differ noticeably in recognition accuracy. Each action was converted into position data and angle data. In addition, we expressed each data by using two visualization methods. The difference in motion characteristics of each subject was easy to distinguish using Motion-Sphere. Next, we showed each angle and position data type's graphs. It was difficult to grasp the consistent characteristics of the same action in the position graph. The position data was considerably affected by human variation.

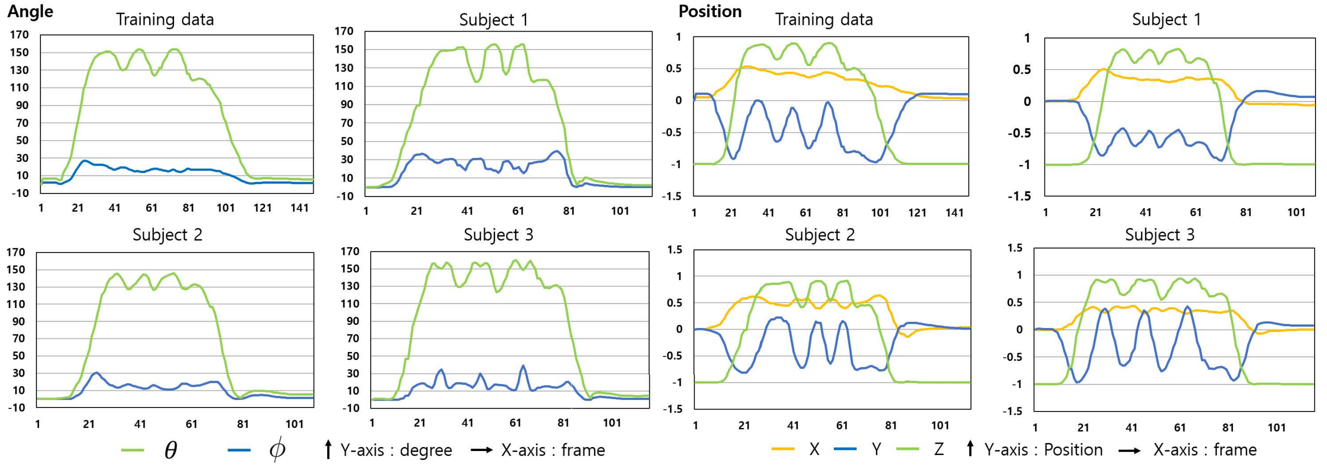

FIGURE 9. Knock action comparison by angle(left) and position(right) graph.

For the Crowbar-right action, the position graphs are different in all subjects. For the Knock action, there is a similar pattern in the z graphs. However, the recognition accuracy is low because the other values work as noisy data. In the angle data, we could observe constant features of the same action by checking the overall graph's shape. For the Crowbar-right action, the θ angle increased, maintained a constant value, and then fell. The ϕ angle had two peak variations when θ remained constant. For the Knock action, the ϕ angle changed with small variations. The θ had three peak regions. Thus, the angle data has the advantage of identifying the uniform patterns within the same action. We can obtain consistent patterns from a new person's angle data. Even if the system lacks the user's information, the system identifies the action consistently. Therefore, adopting angle data as the input data format increased the accuracy.

In this experiment, we tried to analyze the contribution of data types. The HAR system should stably recognize untrained person actions. We virtually simulated such a situation within a unique experiment setting. We confirmed the advantages of the angular data from the experimental results. It is easy to define action's characteristics using angular data. The accuracy of the angle data type is higher than that of position data. We wanted to test the general HAR system in a real-world application. Therefore, we used the recognition system that achieved average performance. Although we used accurate motion data in this experiment, in actual situations it will be more difficult to extract precise motion data. We plan to use authentic data extracted by an actual HAR system. We believe that the HAR system can recognize the poor quality of an action's data if we find the action's consistent definition. We observed that, even if the person changes, the system recognizes the action stably with a consistent pattern of angle data.

V. CONCLUSION

Our research's main aim was the development of a stable HAR system for real-world applications. We mainly concentrated on the training data and the action data type. First, we examined the effects of training data on recognition. We compared the EBR dataset with our DBC dataset. The results showed that the training dataset should contain more detailed actions depending on the specific application. Second, we analyzed data type in terms of stability as motion variation data. We designed a virtual situation that tested an untrained person's data by using the cross-subject protocol. Based on test results, the angle data helped collect general movement patterns. Thus, the recognition system based on the angle data achieved higher accuracy in the cross-subject protocol. To summarize, our paper's main contributions are analyzing the effect of the training data and data types in real-world applications. When we use the HAR system, the training data should include subtle actions, and the action data need to express the general patterns of the action. To exclude the effect of the sensing environment, we used accurate action data. We will use lower-quality action data extracted using a camera-based sensing environment in the future study. In that case, the accuracy of the angle data will decrease. We will enhance our recognizer's performance by a data augmentation method. The augmented motion data can increase the robustness of the recognition system by containing numerous motion variations.

REFERENCES

[1] J. Wang, Y. Chen, S. Hao, X. Peng, and L. Hu, "Deep learning for sensor-based activity recognition: A survey," *Pattern Recognit. Lett.*, vol. 119, pp. 1–3, Mar. 2019.

[2] A. Ullah, K. Muhammad, I. U. Haq, and S. W. Baik, "Action recognition using optimized deep autoencoder and CNN for surveillance data streams of non-stationary environments," *Future Gener. Comput. Syst.*, vol. 96, pp. 386–397, Jul. 2019.

[3] R. Sun, Z. Wang, K. E. Martens, and S. Lewis, "Convolutional 3D attention network for video based freezing of gait recognition," in *Proc. Digit. Image Comput., Techn. Appl. (DICTA)*, Dec. 2018, pp. 1–7, doi: 10.1109/DICTA.2018.8615791.

[4] S. Herath, M. Harandi, and F. Porikli, "Going deeper into action recognition: A survey," *Image Vis. Comput.*, vol. 60, pp. 4–21, Apr. 2017.

[5] R. Poppe, "A survey on vision-based human action recognition," *Image Vis. Comput.*, vol. 28, no. 6, pp. 976–990, Jun. 2010.

[6] J. Han, L. Shao, D. Xu, and J. Shotton, "Enhanced computer vision with Microsoft kinect sensor: A review," *IEEE Trans. Cybern.*, vol. 43, no. 5, pp. 1318–1334, Oct. 2013.

[7] A. Kolahi, M. Hoviattalab, T. Rezaeian, M. Alizadeh, M. Bostan, and H. Mokhtarzadeh, "Design of a marker-based human motion tracking system," *Biomed. Signal Process. Control*, vol. 2, no. 1, pp. 59–67, Jan. 2007.

[8] N. Miller, O. C. Jenkins, M. Kallmann, and M. J. Mataric, "Motion capture from inertial sensing for untethered humanoid teleoperation," in *Proc. 4th IEEE/RAS Int. Conf. Hum. Robots*, vol. 2, Nov. 2004, pp. 547–565.

[9] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-time human pose recognition in parts from single depth images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. CVPR)*, Jun. 2011, pp. 1297–1304.

[10] D. Kim, D.-H. Kim, and K.-C. Kwak, "Classification of *K*-pop dance movements based on skeleton information obtained by a Kinect sensor," *Sensors*, vol. 17, no. 6, p. 1261, Jun. 2017.

[11] H. S. Koppula, R. Gupta, and A. Saxena, "Learning human activities and object affordances from RGB-D videos," *Int. J. Robot. Res.*, vol. 32, no. 8, pp. 951–970, 2013.

[12] A. Balasubramanyam, A. K. Patil, B. Chakravarthi, J. Y. Ryu, and Y. H. Chai, "Motion-sphere: Visual representation of the subtle motion of human joints," *Appl. Sci.*, vol. 10, no. 18, p. 6462, Sep. 2020.

[13] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: A new learning scheme of feedforward neural networks," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, vol. 2, Jul. 2004, pp. 985–990.

[14] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 12, pp. 2247–2253, Dec. 2007.

[15] X. Yang, C. Zhang, and Y. Tian, "Recognizing actions using depth motion maps-based histograms of oriented gradients," in *Proc. 20th ACM Int. Conf. Multimedia*, 2012, pp. 1057–1060.

[16] A. A. Chaaraoui, J. R. Padilla-López, P. Climent-Pérez, and F. Flórez-Revuelta, "Evolutionary joint selection to improve human action recognition with RGB-D devices," *Expert Syst. Appl.*, vol. 41, no. 3, pp. 786–794, Feb. 2014.

[17] A. G. Kirk, J. F. O'Brien, and D. A. Forsyth, "Skeletal parameter estimation from optical motion capture data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2, Jun. 2005, pp. 782–788.

[18] A. K. Patil, A. Balasubramanyam, J. Y. Ryu, B. N. P. Kumar, B. Chakravarthi, and Y. H. Chai, "Fusion of multiple LiDARs and inertial sensors for the real-time pose tracking of human motion," *Sensors*, vol. 20, no. 18, p. 5342, Sep. 2020.

[19] J. Ziegler, H. Kretzschmar, C. Stachniss, G. Grisetti, and W. Burgard, "Accurate human motion capture in large areas by combining IMU- and laser-based people tracking," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2011, pp. 86–91.

[20] A. K. Patil, A. Balasubramanyam, J. Y. Ryu, B. Chakravarthi, and Y. H. Chai, "An open-source platform for human pose estimation and tracking using a heterogeneous multi-sensor system," *Sensors*, vol. 21, no. 7, p. 2340, Mar. 2021.

[21] W. Li, Z. Zhang, and Z. Liu, "Action recognition based on a bag of 3D points," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2010, pp. 9–14.

[22] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Mining actionlet ensemble for action recognition with depth cameras," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1290–1297.

[23] C. Chen, R. Jafari, and N. Kehtarnavaz, "UTD-MHAD: A multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2015, pp. 168–172.

[24] L. Xia, C.-C. Chen, and J. K. Aggarwal, "View invariant human action recognition using histograms of 3D joints," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2012, pp. 20–27.

[25] J. Liu, A. Shahroudy, M. Perez, G. Wang, L.-Y. Duan, and A. C. Kot, "NTU RGB+D 120: A large-scale benchmark for 3D human activity understanding," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 10, pp. 2684–2701, Oct. 2020.

[26] F. Patrona, A. Chatzitofis, D. Zarpalas, and P. Daras, "Motion analysis: Action detection, recognition and evaluation based on motion capture data," *Pattern Recognit.*, vol. 76, pp. 612–622, Apr. 2018.

[27] S. Fothergill, H. Mentis, P. Kohli, and S. Nowozin, "Instructing people for training gestural interactive systems," in *Proc. SIGCHI Conf. Hum. Factors Comput. Syst.*, May 2012, pp. 1737–1746.

[28] M. Jiang, J. Kong, G. Bebis, and H. Huo, "Informative joints based human action recognition using skeleton contexts," *Signal Process., Image Commun.*, vol. 33, pp. 29–40, Apr. 2015.

[29] X. Chen and M. Koskela, "Skeleton-based action recognition with extreme learning machines," *Neurocomputing*, vol. 149, pp. 387–396, Feb. 2015.

[30] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Learning actionlet ensemble for 3D human action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 5, pp. 914–927, May 2014.

[31] R. Qiao, L. Liu, C. Shen, and A. van den Hengel, "Learning discriminative trajectorylet detector sets for accurate skeleton-based action recognition," *Pattern Recognit.*, vol. 66, pp. 202–212, Jun. 2017.

[32] R. Slama, H. Wannous, M. Daoudi, and A. Srivastava, "Accurate 3D action recognition using learning on the Grassmann manifold," *Pattern Recognit.*, vol. 48, no. 2, pp. 556–567, Feb. 2015.

[33] E. Ohn-Bar and M. M. Trivedi, "Joint angles similarities and HOG2 for action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2013, pp. 465–470.

[34] F. Ofli, R. Chaudhry, G. Kurillo, R. Vidal, and R. Bajcsy, "Sequence of the most informative joints (SMIJ): A new representation for human skeletal action recognition," *J. Vis. Commun. Image Represent.*, vol. 25, no. 1, pp. 24–38, Jan. 2014.

[35] A. A. A. Salih and C. Youssef, "Spatiotemporal representation of 3D skeleton joints-based action recognition using modified spherical harmonics," *Pattern Recognit. Lett.*, vol. 83, pp. 32–41, Nov. 2016.

[36] B. Kwon, J. Kim, K. Lee, Y. K. Lee, S. Park, and S. Lee, "Implementation of a virtual training simulator based on 360° multi-view human action recognition," *IEEE Access*, vol. 5, pp. 12496–12511, 2017, doi: 10.1109/ACCESS.2017.2723039.

[37] D. Hodges, "Cyber-enabled burglary of smart homes," *Comput. Secur.*, vol. 110, Nov. 2021, Art. no. 102418.

[38] P. Wang, W. Li, C. Li, and Y. Hou, "Action recognition based on joint trajectory maps with convolutional neural networks," *Knowl.-Based Syst.*, vol. 158, pp. 43–53, Oct. 2018.

[39] C. Li, Y. Hou, P. Wang, and W. Li, "Joint distance maps based action recognition with convolutional neural networks," *IEEE Signal Process. Lett.*, vol. 24, no. 5, pp. 624–628, May 2017.

[40] S. Agahian, F. Negin, and C. Köse, "An efficient human action recognition framework with pose-based spatiotemporal features," *Eng. Sci. Technol., Int. J.*, vol. 23, no. 1, pp. 196–203, Feb. 2020.

[41] R. Zhao, W. Xu, H. Su, and Q. Ji, "Bayesian hierarchical dynamic model for human action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7733–7742.

[42] H. Wang, B. Yu, K. Xia, J. Li, and X. Zuo, "Skeleton edge motion networks for human action recognition," *Neurocomputing*, vol. 423, pp. 1–12, Jan. 2021.

JAEYEONG RYU received the master's degree in computer graphics and virtual reality from Chung-Ang University, South Korea, in 2021, where he is currently pursuing the Ph.D. degree with the Virtual Environments Laboratory. His research interests include virtual reality, HCI, human motion capture systems, and motion recognition.

ASHOK KUMAR PATIL received the Ph.D. degree in CG/VR from the Graduate School of Advanced Imaging Science, Multimedia and Film, Chung-Ang University, Seoul, South Korea, and the Master of Computer Applications degree from Visvesvaraya Technological University, Belagavi, in 2007. He is currently a Postdoctoral Fellow with the Virtual Environment Laboratory, Chung-Ang University. His research interests include computer graphics, virtual reality, interactive systems, HCI, and automation in 3D reconstruction.

BHARATESH CHAKRAVARTHI received the Bachelor of Engineering degree in information science and the Master of Technology degree in computer networks and engineering from Visvesvaraya Technological University, Karnataka, India, in 2011 and 2013, respectively. He is currently pursuing the Ph.D. degree in computer graphics and virtual reality from the Graduate School of Advanced Imaging Science, Multimedia and Film, Chung Ang University, Seoul, South Korea. His research interests include human motion capture systems, human motion visualization, sensor, computer graphics, and virtual reality.

ADITHYA BALASUBRAMANYAM received the Ph.D. degree in computer graphics and virtual reality from Chung-Ang University, Seoul, South Korea. He is currently a Faculty Member with the Department of Computer Science and Engineering, PES University, Bengaluru. His research interests include human motion tracking, virtual reality, and robotics.

SOUNGSILL PARK received the master's degree in computer graphics and virtual reality from Chung-Ang University, South Korea, in 2016, where she is currently pursuing the Ph.D. degree with the Virtual Environments Laboratory. Her research interests include virtual reality and human action recognition.

YOUNGHO CHAI received the M.S. degree in mechanical engineering from SUNY Buffalo and the Ph.D. degree in mechanical engineering from Iowa State University, in 1997. From 2006 to 2007, he was with Louisiana Immersive Technology Enterprise (LITE), University of Louisiana at Lafayette, USA. He is currently a Professor with the Graduate School of Advanced Imaging Science, Multimedia, and Film, Chung-Ang University, Seoul, South Korea, where he leads the Virtual Environments Laboratory. His research interests include spatial sketching, HCI, HAR, and motion recognition.