Can Ensemble of Classifiers Provide Better Recognition Results in Packaging Activity?

A. H. M. Nazmus Sakib, Promit Basak, Syed Doha Uddin, Shahamat Mustavi Tasin, and Md Atiqur Rahman Ahad

Abstract Skeleton-based motion capture (MoCap) systems have been widely used in the game and film industry for mimicking complex human actions for a long time. MoCap data has also proved its effectiveness in human activity recognition tasks. However, it is a quite challenging task for smaller datasets. The lack of such data for industrial activities further adds to the difficulties. In this work, we have proposed an ensemble-based machine learning methodology that is targeted to work better on MoCap datasets. The experiments have been performed on the MoCap data given in the Bento Packaging Activity Recognition Challenge 2021. *Bento* is a Japanese word that resembles *lunch-box*. Upon processing the raw MoCap data at first, we have achieved an astonishing accuracy of 98% on tenfold cross-Validation and 82% on leave-one-out cross-validation by using the proposed ensemble model.

1 Introduction

Human activity recognition has been one of the major concentrations for researchers for over a decade. Previously, human activity recognition tasks only used data from geospatial sensors such as accelerometers, gyroscopes, GPS sensors [1]. But in the last few years, skeleton-based human action recognition (SHAR) became quite popular because of its better performance and accuracy [2–5]. In SHAR, the human skeleton is typically represented by a set of body markers which are tracked by several specialized cameras. In such work, computer vision or sensor data can also be used. In the case of sensor data, the use of motion capture, kinematic sensors, etc. are prominent [6]. Although skeleton-based data is already being used widely in many cases, such applications are quite rare in fields such as packaging, cooking [7], nurse care [8]. Among these fields, packaging activity recognition can be very effective in

A. H. M. Nazmus Sakib · P. Basak · S. Doha Uddin · S. Mustavi Tasin
Electrical and Electronic Engineering, University of Dhaka, Dhaka, Bangladesh

M. A. R. Ahad (✉)
University of East London, London, UK
e-mail: atiqahad@du.ac.bd; atiqahad@yahoo.com

M. A. R. Ahad et al. (eds.), *Sensor- and Video-Based Activity and Behavior Computing*,
Smart Innovation, Systems and Technologies 291,
https://doi.org/10.1007/978-981-19-0361-8_10

the industrial arena and can solve multi-modal problems like industrial automation, quality assessment, and reducing errors.

Packaging activity recognition is a relatively newer field of SHAR. Hence, scarcity of data and lack of previous examples are some of the main problems of this task. Usually, in such tasks, items are put on a conveyor belt and packaged by a subject. The items to be put on the conveyor belt are dictated by the company. Hence, the location of the item on the belt is also important as it produces new scenarios for different positions. The size of the dataset to perform such experiments is another important factor because small datasets often limit the scope of work. In this paper, our team *Nirban* proposes an approach to solve these issues and detect such activities in the Bento Packaging Activity Recognition Challenge 2021 [9, 10].

The dataset used in this work is based on Bento box packaging activities. Bento is a single-serving lunch-box originated in Japan. The dataset contains motion capture data with 13 body markers with no previous preprocessing. The raw motion capture data is first preprocessed, and then a total of 2400 features are extracted. Furthermore, feature selection is used to select the best 396 features based on “mean decrease in impurity” and chi-square score. Then, the processed data is trained on several classical machine learning models, and their performances are evaluated using tenfold cross-validation (CV) and leave-one-out cross-validation (LOOCV). Lastly, an ensemble of the best five models is done to generate predictions on the test data. Deep learning (DL) methods were also considered, in which case raw data was fed to one-dimensional CNN, LSTM, and bidirectional LSTM. The results of all the approaches and models are included in this paper.

The rest of the paper is organized as follows: Sect. 2 describes the previous works that are relevant to the approach described here. Section 3 provides a detailed description of the dataset, including its settings and challenges. Section 4 entails the detailed methodology used in this work. Section 5 describes the results and analysis of the results, including the approach, as well as the future scopes of this work. Finally, the conclusion is drawn in Sect. 6.

2 Related Work

Several pieces of research regarding SHAR have been carried out where the dataset had motion capture data. Picard et al. [11] used motion capture data to recognize cooking activities. The method achieved a score of 95% on cross-validation of the training set. In this work, a subject has been visualized as a stickman using the MoCap data. For temporal information to be taken into consideration, an HMM model was used in post-processing to get a better result. It should be noted that the dataset had few wrong labels, which were manually labeled here, and data was shuffled which was also ordered before training. It helped them reach such a high accuracy. For different classes, two specialized classifiers were used, and their results were merged.

Image-based approaches have been also observed in the industrial packing process of range hood (meaning kitchen chimney) [12]. In this case, local image-directed graph neural network (LI-DGNN) is used on a set of different types of data. The dataset includes RGB videos, 3D skeleton data extracted by pose estimation algorithm AlphaPose, local images, and bounding-box data of accessories. However, since it uses local images from video frames, it is subjected to object occlusion and viewpoint changes if used solely. Also, as the items are needed to be tracked continuously, it causes a bottleneck in using local images. As a result, a combination of local images and other sensor data is required.

An important observation regarding SHAR works is the size of the datasets. In most cases, the dataset is large enough to experiment with deep learning approaches. Deep learning methods are expected to outperform classical machine learning models with hand-crafted features [13]. In this case, a large dataset is advantageous for a data-driven method. However, such an approach is not expected to work well on smaller datasets as given in the Bento Packaging Activity Recognition Challenge 2021.

3 Dataset

3.1 Data Collection Setup

In any activity recognition challenge, the environment in which data is collected is very crucial. For the Bento Packaging Activity Recognition Challenge 2021, data is collected in the Smart Life Care Unit of the Kyushu Institute of Technology in Japan. Motion Analysis Company [14] has provided the necessary instruments to collect motion capture data. The setup consists of 29 different body markers, 20 infrared cameras to track those markers, and a conveyor belt, where the Bento boxes are passed. Though there were 29 different body markers initially, data for only 13 markers of the upper body is provided for this challenge. The marker positions are shown in Fig. 1.

The data is collected from 4 subjects aged from 20 to 40. While collecting data, empty Bento boxes are passed to the subject using a conveyor belt, and the subject has to put three kinds of foods in the box. The data collection setup is given in Fig. 2, where a subject is taking food and putting it in the Bento box. The face is covered with a white rectangular mask to protect privacy.

3.2 Dataset Description

The dataset for the Bento Packaging Activity Recognition Challenge 2021 consists of activities from five different scenarios necessary for packaging a Bento box. For

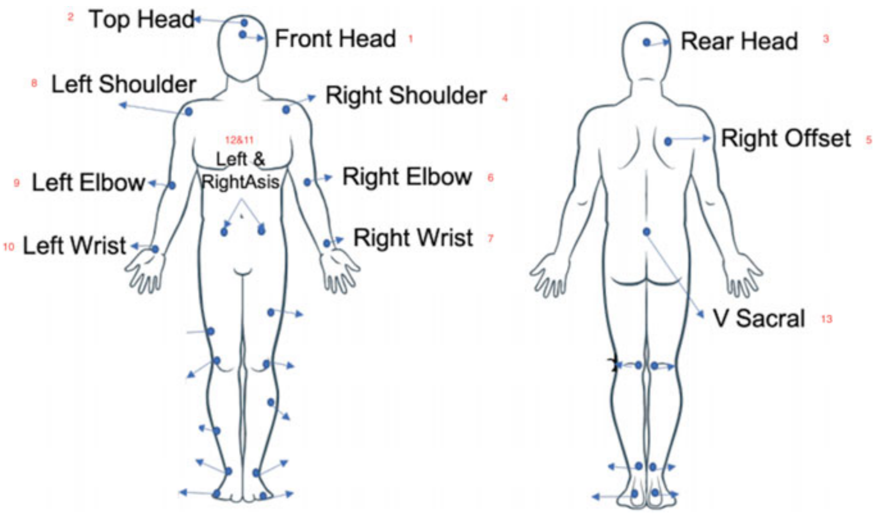


Fig. 1 Position of the markers (*Source* https://abc-research.github.io/bento2021/data/)

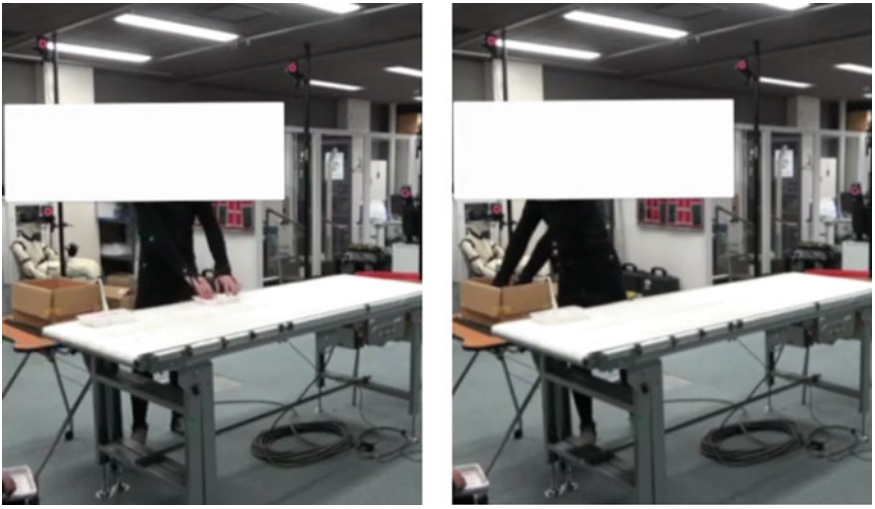


Fig. 2 Data collection setup. A subject is putting items in a Bento box, which is on the white conveyer belt in the middle. At left-mid, the rectangular box is used to cover the subject's face to retain privacy (*Source* https://youtu.be/mQgCaCjC7fI)

each scenario, the activity is done in two different patterns which are inward and outward. The name and label for each activity are listed in Table 1.

The provided data contains three-dimensional coordinates for each of the body markers sampled at a frequency 100 Hz. Each subject has performed each activity approximately five times, and the duration of each activity is between 50 and 70 s. There are a total of 151 training files and 50 test files where each file represents a single activity. From the subject-wise data distributions shown in Fig. 3, it is evident that the dataset is well balanced and each of the subjects has done almost an equal number of different activities.

Table 1 Activity names and corresponding labels

Activity name	Label
Normal (inward)	1
Normal (outward)	2
Forgot to put ingredients (inward)	3
Forgot to put ingredients (outward)	4
Failed to put ingredients (inward)	5
Failed to put ingredients (outward)	6
Turn over bento box (inward)	7
Turn over Bento box (outward)	8
Fix/rearranging ingredients (inward)	9
Fix/rearranging ingredients (outward)	10

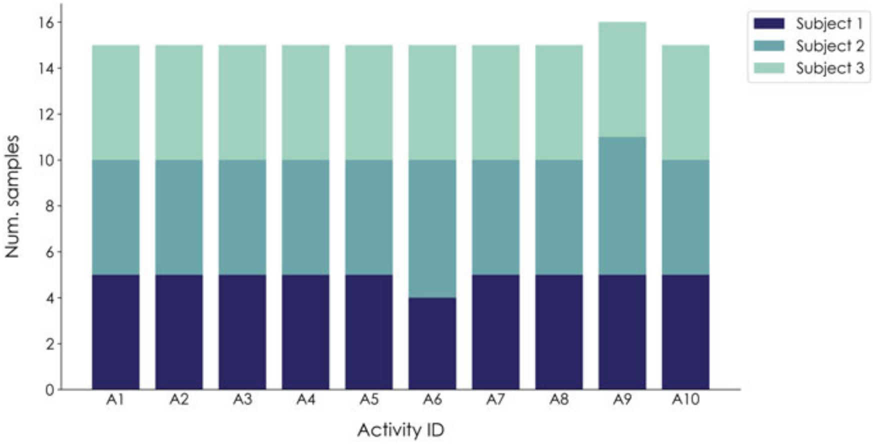


Fig. 3 Distribution of samples for every class (A1~A10). Data of three subjects (as per the training set) is shown

3.3 Dataset Challenges

In most cases, a real-life data collection setup has some inevitable inconsistencies, which make it challenging to work on. The dataset given in this challenge is not free from this issue too. The biggest challenge of this dataset is the small amount of data. There is only a total of 151 instances in the training set which is extremely low considering the complexity of each activity. This problem makes it very hard to use deep learning models that require large datasets to perform well [15]. Another difficulty of the dataset is shown in Fig. 4, which compares the average execution time of each activity for different subjects. It is obvious that different subjects have done the activities differently. Subject 2 has taken a significantly longer period to execute the activities in comparison with Subjects 1 and 3. This problem is more evident for Activities 7, 8, and 9. As the test set contains actions from a different subject who is not present in the training set, this cross-subject inconsistency is likely to take a toll on the overall performance of the model.

However, there are some other problems in the dataset. The data is provided in raw *.csv files rather than any conventional motion capture data format such as *.bvh, *.htr *.c3d, *.asf [16]. The base positions of the body joints are not provided too. As a result, many important features can not be extracted properly from the files. The setting of data collection is very complex which caused incorrect marker labeling, missing data, and unwanted noises which offered further challenges. Of the 29 markers, data from only 13 markers of the upper body was given. Some of the activities are easily separable if lower body marker data is provided. We have addressed almost every issue in our work, which will be covered in the next sections.

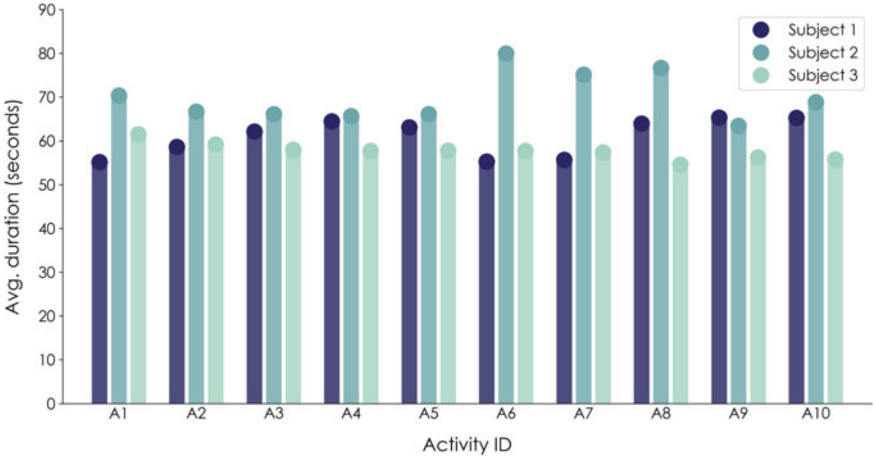


Fig. 4 Subject-wise average activity distribution for three subjects. Distribution for test subject is unknown

4 Methodology

4.1 Preprocessing

fs The most prominent challenge of this dataset is the low number of instances in the training data. To minimize the problem of data scarcity, we have divided each data into multiple overlapped segments of 20–40 s. Smaller segments increase the number of instances sacrificing the global trend, while larger segments decrease the number of instances retaining the global trend. Hence, we have treated the segment size as one of the hyperparameters and have tuned it to the perfect value. A similar approach is taken on the overlapping rate of two consecutive segments. After completing the preprocessing, we were able to increase the number of instances to a range of 300–600 for different combinations of segment size and overlapping rate.

Also, there are some missing values in the dataset. We have interpolated them linearly instead of imputation as the dataset has been resampled at a constant frequency. We have extracted features for each segment which will be described in the following subsection.

4.2 Stream and Feature Extraction

In this dataset, only the cartesian coordinates of 13 upper body joints are given (Fig. 5). So, for each activity, there is a sequence of positions in three axes for all 13 markers. For describing purposes, we will call each of the temporal sequences a stream. This position stream is not enough to describe each activity. So, we have

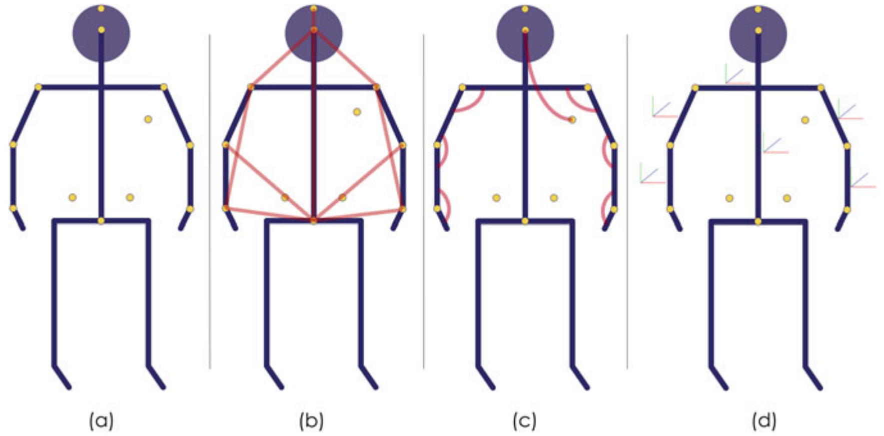


Fig. 5 Stream extraction steps: **a** Skeleton model, **b** Extraction of distance streams, **c** Extraction of joint angle streams, and **d** Extraction of planer angle streams

differentiated it repeatedly to get the speed, acceleration, and jerk streams. In Fig. 5, stream extraction steps are mentioned.

In real life, we move our certain limbs to execute any action, and the distances between certain body joints, d, are crucial for detecting any action [2]. From this point of view, we have calculated distances between selected body joints (i.e., the distance between wrist and shoulder, between v-sacral and elbow, between front head and elbows, between the wrists) to create the distance stream. Each distance signifies a separate concept. For example, the distance between the v-sacral and front head helps to determine if the person has bent his/her head or not. The distance is defined as,

$$d = \sqrt{x^2 + y^2 + z^2} \tag{1}$$

For different activities, the angle between three consecutive joints, θ, and the orientation of the selected bones (i.e., forehand, hand) should be very important. Hence, we have extracted selected joint angle streams and planar angles for bone streams too. In both cases, we have differentiated each stream to get the angular speed streams, according to the following expression,

$$\theta = \arccos\left(\frac{\mathbf{v_1}}{\|v_1\|} \cdot \frac{\mathbf{v_2}}{\|v_2\|}\right) \tag{2}$$

We have synthesized a total of 218 streams after the stream extraction process. The next process is different for RNN-based models and traditional machine learning models. RNN-based deep learning models like LSTM can take each stream directly as the input of the network as well-crafted deep learning networks can learn features from data on their own. Hand-crafted feature extraction is unnecessary for these models. But, the number of streams is too much for the dataset size. For this reason, we have selected the most important 40 streams for LSTM to train on.

We have observed that the deep learning models performed very poorly because the dataset was so small. So, we have constructed a separate feature extraction pipeline for traditional machine learning models. From each of the selected streams, we have extracted the basic frequency-domain features (i.e., median, skew, kurtosis, energy) apart from some statistical features (i.e., median, min, max, standard deviation).

After the execution of the feature extraction pipeline, the feature set became quite large compared to the dataset. So, we had to remove a considerable portion of the feature set to prevent overfitting. We have used the mean decrease in impurity [17] and chi-square techniques [15] to select the most significant 496 features for the later workflow.

4.3 Model Selection and Post-processing

Even after taking a much smaller set of features through the feature selection methods, the number of features remains quite high compared to the dataset size. After preprocessing, the highest number of instances we have produced is less than 600, where the number of features is already as many as 496. This data-to-feature ratio will highly likely lead to overfitting. So, we have proposed a model ensembling system to solve this problem (Fig. 6).

First, we have divided the feature set into 13 overlapped feature subsets each of which contained approximately 150–250 features. For each model, we have trained on different feature subsets and evaluated its result on both tenfold CV and LOOCV. We have selected the top five models based on both the evaluation scores and added a majority voting layer on top of each of the models' predictions. The intuition behind the proposed ensemble system is that the reduced feature set will make the models less prone to overfitting, and majority voting will combine all predictions and give us a final prediction that considers the full feature set [18].

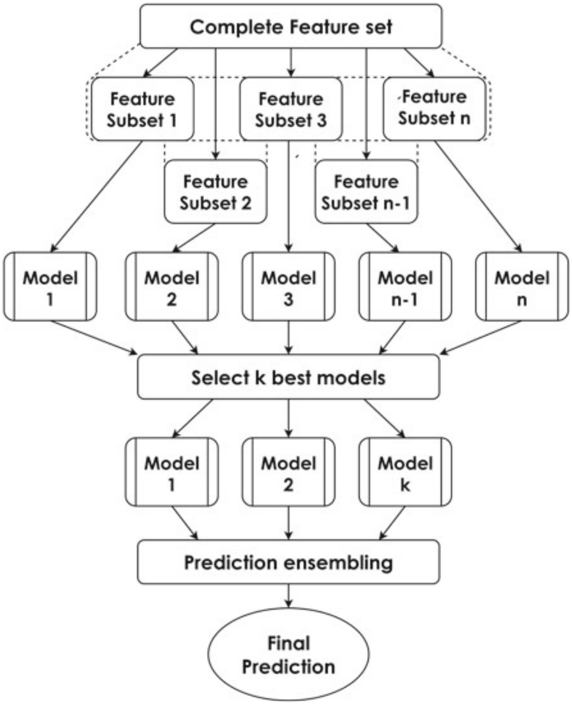


Fig. 6 Our propesed ensemble-based framework

5 Results and Analysis

We have used different tuned models and evaluated them based on tenfold CV and LOOCV. We have found the extra trees classifier to perform best on our framework. The detailed results are portrayed in Table 2. The confusion matrices of the best model found after LOOCV and tenfold CV are depicted in Fig. 7.

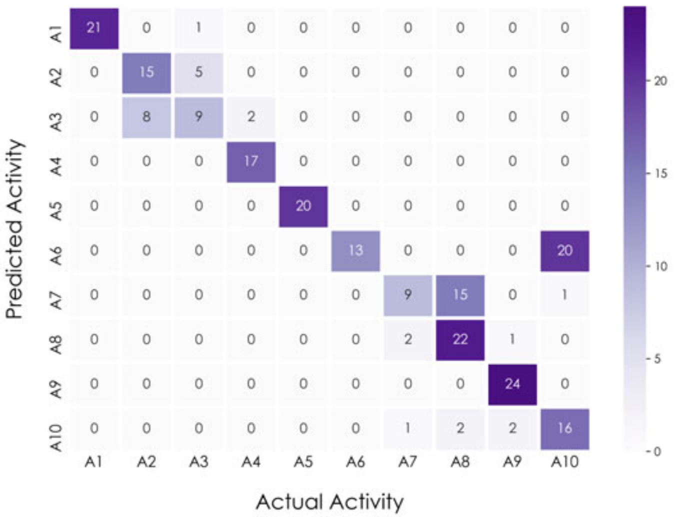
The highest accuracy we have achieved in this work is 98% for tenfold CV and 82% for LOOCV. Though the result we obtained is not perfect, this is a very competitive result considering the dataset challenges. There are several reasons behind it. As we can see from Table 2, the evaluation on LOOCV is pretty much less than that of the tenfold CV score. The reason behind it is the lower number of samples for each activity. Each subject has only five samples for each activity which is too low for generalizing on another subject, and this leads to a significantly lower score on LOOCV.

Also from Table 2, we can see that some models have done significantly poorer than other models. LSTM has performed the worst because of the small dataset size. Deep learning models generally need a lot of data to obtain a generalized performance on data. In this case, the dataset size is so small that LSTM has performed even lower than the baseline model, which is the naive Bayes classifier. Gradient boosting models, XGBoost, and LightGBM also suffered a lot from this problem and do not perform very well.

If we look at the confusion matrix for LOOCV in Fig. 7a, we can notice that the model can not differentiate between Classes 2 and 3, 7 and 8, and 6 and 10. It is because different subjects carried out the activities differently to some extent. This problem is also evident in Fig. 4, which depicts the average activity execution time for different subjects. For example, for Activity 6, Subject 1 and Subject 3 took 60 s on average to perform, but Subject 2 took more than 80 s to perform the same task. This problem is reflected in the confusion matrix as we can see that the model was severely confused between Activity 6 and Activity 10. Also, Activities 2 and 3 are typically distinguishable, but there are some confusions observed in the confusion

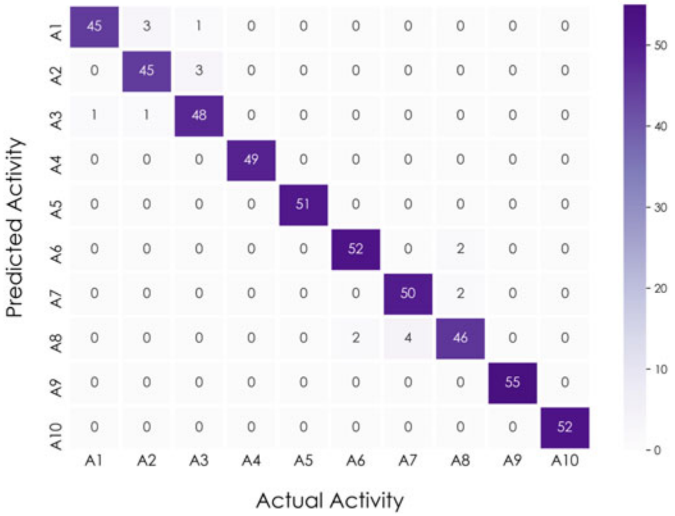
Table 2 Accuracy of different models

Model	Tenfold CV accuracy	LOOCV accuracy
Naive Bayes	0.84	0.64
Support vector machine (SVM)	0.92	0.73
Random forest classifier (RFC)	0.96	0.77
Extra trees classifier (ETC)	0.96	0.78
LightGBM	0.94	0.74
XGBoost	0.88	0.67
Long short-term memory (LSTM)	0.61	0.37
Ensemble of four RFC models and one ETC model	0.98	0.82



	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10
A1	21	0	1	0	0	0	0	0	0	0
A2	0	15	5	0	0	0	0	0	0	0
A3	0	8	9	2	0	0	0	0	0	0
A4	0	0	0	17	0	0	0	0	0	0
A5	0	0	0	0	20	0	0	0	0	0
A6	0	0	0	0	0	13	0	0	0	20
A7	0	0	0	0	0	0	9	15	0	1
A8	0	0	0	0	0	0	2	22	1	0
A9	0	0	0	0	0	0	0	0	24	0
A10	0	0	0	0	0	0	1	2	2	16

(a) Confusion matrix for the best model after LOOCV.



	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10
A1	45	3	1	0	0	0	0	0	0	0
A2	0	45	3	0	0	0	0	0	0	0
A3	1	1	48	0	0	0	0	0	0	0
A4	0	0	0	49	0	0	0	0	0	0
A5	0	0	0	0	51	0	0	0	0	0
A6	0	0	0	0	0	52	0	2	0	0
A7	0	0	0	0	0	0	50	2	0	0
A8	0	0	0	0	0	2	4	46	0	0
A9	0	0	0	0	0	0	0	0	55	0
A10	0	0	0	0	0	0	0	0	0	52

(b) Confusion matrix for the best model after 10-fold CV.

Fig. 7 Confusion matrices

matrix for LOOCV. On the other hand, Activities 7 and 8 are similar, but confusions are observed in this case too. Video data of these activities might help to solve these issues in the future. Despite different challenges and complications of the dataset, our proposed procedure has managed to achieve a quite promising result.

6 Conclusion

In this paper, we have provided a method to tackle the challenges of activity recognition in an industrial setting. Though human activity recognition is a very popular field and a wide variety of work has been done, our work still manages to provide a solution in a less explored arena of this field. After comparing various methods used in previous works by applying them to our dataset, we have decided to use a hand-crafted feature-based solution for our final approach. We have calculated various streams such as speed, joint angle, marker distance from the given data. Furthermore, we have used segmentation with overlap to increase the amount of the data. After that, we have extracted statistical features from each stream. The features are then used to train different machine learning models. After tuning the models and evaluating them using LOOCV and tenfold CV methods, five best-performing models (four random forest classifiers and one extra tree classifier) were selected. The models are used to make predictions on different segments generated from the files of the test dataset, and a majority voting system among the models generates the final predictions.

Our method provides a good amount of precision, but further improvement is still possible. We have experimented with quaternion data generated from provided three-dimensional coordinates but it could not manage to obtain significant improvement through its use. Finding a system to incorporate this data might provide better accuracy. We have also explored different deep learning methods (i.e., temporal convolutional network, LSTM-based encoder-decoder, bidirectional LSTM-based network), but they perform poorly. We believe that the reason for this is the small size of the dataset. Deep learning approaches have the potential to perform better than machine learning approaches, provided that more data is collected. It will also be able to provide end-to-end solutions in contrast to our hand-crafted feature-based solution, which will streamline its integration in the industry. Thus, the collection of more data and exploring deep learning approaches on the data should be strongly considered for future works.

Appendix

See Table 3.

Table 3 Miscellanious information

Information heading	Description
Used sensor modalities	Motion capture (MoCap)
Features used	As described in Sect. 4.2
Programming language	Python 3.8
Packages used	Pandas, NumPy, SciPy, Scikit-learn, XGBoost, TensorFlow, Keras
Machine specifications	Intel(R) Core(TM) i5-7200U CPU @ 2.50GHz, 8 GB RAM
Training and testing time	Time to process and train on full training data—12 min Time to predict on full test data—3 min

References

1. Óscar D. Lara, Labrador, M.A.: A survey on human activity recognition using wearable sensors. IEEE Commun. Surveys Tutorials **15**, 1192–1209 (2013). https://doi.org/10.1109/SURV.2012.110112.00192
2. Cippitelli, E., Gasparrini, S., Gambi, E., Spinsante, S.: A human activity recognition system using skeleton data from rgbd sensors. Comput. Intell. Neurosc. **2016** (2016). https://doi.org/10.1155/2016/4351435
3. Núñez, J.C., Cabido, R., Pantrigo, J.J., Montemayor, A.S., Vélez, J.F.: Convolutional neural networks and long short-term memory for skeleton-based human activity and hand gesture recognition. Pattern Recogn. **76**, 80–94 (2018). https://doi.org/10.1016/J.PATCOG.2017.10.033
4. Sarker, S., Rahman, S., Hossain, T., Ahmed, S.F., Jamal, L., Ahad, M.A.R.: Skeleton-Based Activity Recognition: Preprocessing and Approaches, pp. 43–81. Springer International Publishing (2021). https://doi.org/10.1007/978-3-030-68590-4_2
5. Zhu, W., Lan, C., Xing, J., Zeng, W., Li, Y., Shen, L., Xie, X.: Co-occurrence feature learning for skeleton based action recognition using regularized deep lstm networks. In: Proceedings of the AAAI Conference on Artificial Intelligence **30** (2016). https://ojs.aaai.org/index.php/AAAI/article/view/10451
6. Ahad, M.A.R., Ahmed, M., Antar, A.D., Makihara, Y., Yagi, Y.: Action recognition using kinematics posture feature on 3d skeleton joint locations. Pattern Recogn. Lett. **145**, 216–224 (2021). https://doi.org/10.1016/J.PATREC.2021.02.013
7. Cooking activity recognition challenge. https://abc-research.github.io/cook2020/ (2020). Accessed: 21 Aug 2021
8. Basak, P., Tasin, S.M., Tapotee, M.I., Sheikh, M.M., Sakib, A.H., Baray, S.B., Ahad, M.A.: Complex nurse care activity recognition using statistical features. In: UbiComp/ISWC 2020 Adjunct—Proceedings of the 2020 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2020 ACM International Symposium on Wearable Computers, pp. 384–389 (2020). https://doi.org/10.1145/3410530.3414338
9. Adachi, K., Alia, S.S., Nahid, N., Kaneko, H., Lago, P., Inoue, S.: Summary of the bento packaging activity recognition challenge. In: The 3rd International Conference on Activity and Behavior Computing (ABC2021) (2021)
10. Alia, S.S., Adachi, K., Nahid, N., Kaneko, H., Lago, P., Inoue, S.: Bento packaging activity recognition challenge (2021). https://doi.org/10.21227/cwhs-t440
11. Picard, C., Janko, V., Reščič, N., Gjoreski, M., Luštrek, M.: Identification of cooking preparation using motion capture data: A submission to the cooking activity recognition challenge.

Smart Innovation, Syst. Technol. **199**, 103–113 (2021). https://doi.org/10.1007/978-981-15-8269-1_9

12. Chen, Z., Hu, H., Li, Z., Qi, X., Zhang, H., Hu, H., Chang, V.: Skeleton-based action recognition for industrial packing process. In: IoTBDS 2020—Proceedings of the 5th International Conference on Internet of Things, Big Data and Security pp. 36–45 (2020). https://doi.org/10.5220/0009340800360045
13. Hossain, T., Sarker, S., Rahman, S., Ahad, M.A.R.: Skeleton-based human action recognition on large-scale datasets. Intell. Syst. Ref. Libr. **207**, 125–146 (2021). https://doi.org/10.1007/978-3-030-75490-7_5
14. Motion capture analysis software. https://motionanalysis.com/movement-analysis/ (2021). Accessed: 21 Aug 2021
15. Suto, J., Oniga, S., Sitar, P.P.: Comparison of wrapper and filter feature selection algorithms on human activity recognition. In: 2016 6th International Conference on Computers Communications and Control, ICCCC 2016 pp. 124–129 (2016). https://doi.org/10.1109/ICCCC.2016.7496749
16. Meredith, M., Maddock, S.: Motion capture file formats explained. Production (2001)
17. Nguyen, T.T., Huang, J.Z., Nguyen, T.T.: Unbiased feature selection in learning random forests for high-dimensional data. Scientific World J. **2015** (2015). https://doi.org/10.1155/2015/471371
18. Bayat, A., Pomplun, M., Tran, D.A.: A study on human activity recognition using accelerometer data from smartphones. Proced. Comput. Sci. **34**, 450–457 (2014). https://doi.org/10.1016/J.PROCS.2014.07.009