Summary of the *Cooking Activity Recognition Challenge*

Sayeda Shamma Alia, Paula Lago, Shingo Takeda, Kohei Adachi, Brahim Benaissa, M.A.R. Ahad and Sozo Inoue

Abstract *Cooking Activity Recognition Challenge* [1] is organized as a part of ABC2020 [2]. In this work, we analyze and summarize the approaches of submissions of the *Challenge*. A dataset consisting of macro and micro activities, collected in *Cooking scenario* were opened to the public with a goal of recognizing both of these activities. The participant teams used the dataset and submitted their predictions of test data which was released on March 1st, 2020. The submission of the teams were evaluated rigorously and the winning team achieved about 35.4% averaged accuracy for macro and micro activities.

1 Introduction

The combination of the Internet of Things (IoT) with Artificial Intelligence (AI) is giving rise to services and applications for personalized health care and monitoring.

Sayeda Shamma Alia
Kyushu Institute of Technology, e-mail: alia@sozolab.jp

Paula Lago
Kyushu Institute of Technology, e-mail: paula@mns.kyutech.ac.jp

Shingo Takeda
Kyushu Institute of Technology, e-mail: takeda@sozolab.jp

Kohei Adachi
Kyushu Institute of Technology, e-mail: adachi@sozolab.jp

Brahim Benaissa
Kyushu Institute of Technology, e-mail: brahim@sozolab.jp

M.A.R. Ahad
University of Dhaka, e-mail: atiqahad@du.ac.bd

Sozo Inoue
Kyushu Institute of Technology, e-mail: sozo@brain.kyutech.ac.jp

Among those services, monitoring at home has sparked attention for its impact in elder care services that allow them to *Age in Place* [1] while ensuring their safety. These services provide awareness of what the user is doing to alert in case of emergencies. For example, by fall detection [3] immediate medical service can be dispatched to them. The future possibilities of using these services in various fields are numerous. An example can be, people who need special attention like Autism spectrum disorder [4], Dementia [5], Parkinson's disease [6] patients can be monitored at home while providing tailored care to them based on their need. Another example is automatic record creation for nurses in hospitals.

One important use case for the elderly is cooking monitoring. As the activity of cooking is a strong indicator of cognitive health and independent living ability, and it opens up the door for monitoring nutrition. Cooking is a complex activity, usually made up of several smaller activities like "taking from the fridge", "washing the food", "mix in the bowl". Recognizing such steps can have several advantages. For instance, in the scenario of an elder living alone, recognizing the steps can be used to remind them of a missing step, or to ensure a healthy diet is being followed. In the scenario of the nursing record, recognizing the steps can be useful for care quality assessment, or for ensuring that safety protocols have been followed, like washing the hands at the proper moments.

Cooking is a complex activity. Monitoring studies use several sensors, embedded in the environment such as temperature and motion sensors, as well as electric consumption sensors. However, such installations might be costly and difficult to maintain. Therefore, in this *challenge*, we explored the possibility of cooking activity monitoring with wearable (smart watch) and smartphone sensors. Which are cheaper and already available at homes. We collected a dataset [8] consisting of three recipes and nine steps (CUT, TAKE, MIX, ADD, OTHER, POUR, OPEN, PEEL, WASH).

While the dataset was collected in laboratory setup, this dataset is considered to make the initial evaluation of the cooking activity and to get a sense of its complexity. For the goal of automatic recognition of the recipes that are being prepared, and to monitor the steps followed to make them. Current activity recognition systems focus on recognizing either the complex label (macro activity) or the small steps (micro activities) but their combined recognition is critical for real life application analysis. In fact, in a nursing scenario, washing the hands after taking blood is very different than doing it before, as it is mandatory. Thus, this *challenge* aimed at the recognition of the macro and micro activities taking place during cooking sessions.

In this paper, we provide an overview of the submissions received to the *challenge*, analyzing their approaches as well as meta data and highlighting the lessons learned.

[1] https://en.wikipedia.org/wiki/Aging_in_place

2 Dataset Description

The dataset is collected in a setting where cooking activities are performed. Each of the subjects are instructed to cook three foods following specific recipes. Cooking process of these three foods are considered as three different activities. In this section, details of the activities of this dataset will be described as well as data collection environment and the used sensors, will be reported.

2.1 Activities Collected

The dataset used for this *challenge* consists of activities and actions associated with cooking. Actions are named as **Micro activities** and activities are named as **Macro activities**. There are three macro activities and 9 micro activities. Each macro activity is consist of multiple micro activities. Details of each macro activity is given below.

- **CEREAL:** TAKE , OPEN , CUT , PEEL , OTHER, PUT
- **FRUITSALAD:** TAKE , ADD , MIX , CUT , PEEL , OTHER, PUT
- **SANDWICH:** TAKE , CUT , OTHER, WASH , PUT

As we can see, the macro activities have many similar micro activities which are done in slightly different ways. This increases the difficulty level for correctly detecting these activities. We can see the reflection of this statement in the confusion matrix of the winning team (Figure 10).

Data is divided into 30 second segments, macro and micro labels are given for each segment. Most of the time micro activities are done in less than 30 seconds, so multiple labels for micro activities are observed in most of the segments. Number of samples for each of the activities are shown in Table 1.

Table 1: Number of samples for training and testing

	Classes	Training	Testing
Macro	CEREAL	73	26
	FRUITSALAD	102	38
	SANDWICH	113	35
Micro	TAKE	134	46
	ADD	18	6
	MIX	19	4
	OPEN	23	6
	CUT	99	31
	PEEL	96	36
	OTHER	74	34
	WASH	30	10
	PUT	114	46

2.2 Experimental Settings and Sensor Modalities

The data collection experiment was conducted in *Smart Life Care Unit* of the Kyushu Institute of Technology in Japan. Four subjects participated during data collection and there was no overlap between the subjects. The experiment was conducted in a controlled environment where the steps are pre-defined for the subjects. They had to prepare three types of foods following the defined steps.

The data was collected using smartphone, smart watches, motion capture system and open pose. Only the first three sensors are open to the public for this *challenge*. The details of each sensors are given below.

Motion Capture: Motion capture system from Motion Analysis Company [9] is used for this experiment. It has 29 body markers. The places of markers in the body are shown in Figure 1. 16 infrared cameras are used to track the markers.

Accelerometer sensor: 2 smart phones placed in right arm and left hip, 2 smart watches placed in both of the wrist of a subject. The used smartwatches are TicWatch E. Samsung Galaxy S9 SCV38 and Huawei P20 Lite smartphone are used in left hip and right arm consecutively.

Open Pose [10]: It is a real-time 2D open-source pose detection system. It can detect 135 key points of human body from a single image. But during this experiment, among the key points, only the marker points of motion capture are used in open pose.

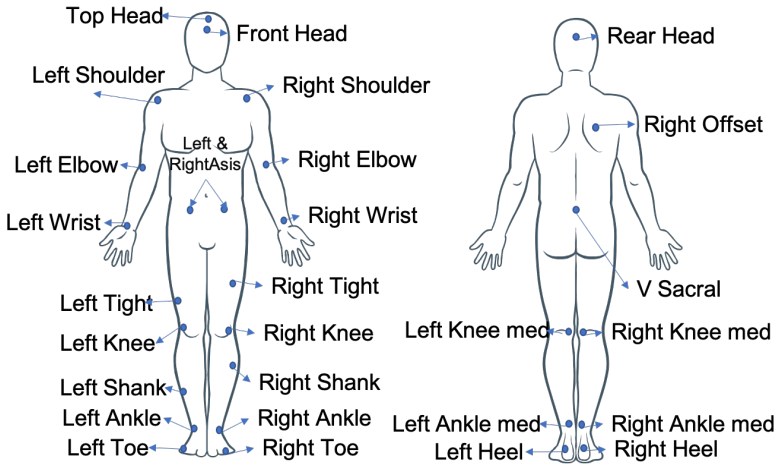


Fig. 1: Motion capture markers used in this dataset

2.3 Data Format

The data has been separated into training data and test data. Training data contains data from 3 subjects and test data contains the fourth subject's data. Each recording has been segmented into 30-second segments as mentioned earlier. Each segment was assigned a random identifier, so the order of the segments is unknown. Each of these segments are recorded in one csv file individually. Data collected by each of the sensors are represented by one folder. One row of the each file contains the file name, the macro activity and the micro activities are all separated by commas. The structure of a folder is shown in Figure 2.

Fig. 2: Folder structure for the dataset

3 Challenge Tasks and Results

The goal of the *Cooking Activity Recognition Challenge* is to recognize both the macro and the micro activities. The *challenge* provides a training and test data. The participants are asked to predict the macro and micro labels for the test data. Evaluation is done based on the submitted labels for both of the activities separately.

Initially 78 unique teams registered, but 9 teams submitted in the final *challenge*. This large difference in number can be caused by few reasons:

- Global pandemic of COVID19
- Not enough time
- Could not get high accuracy

The reasons are obtained through talking with several teams that could not submit for the final stage of the *challenge*.

3.1 Evaluation Metric

To evaluate the submission, accuracy is calculated for both macro and micro labels individually. Accuracy of the macro labels is obtained using Equation 1.

$$Accuracy(A_{ma}) = \frac{CorrectlyPredictedSamples}{TotalSamples} \tag{1}$$

For micro activity, accuracy is calculated using the multilabel accuracy formula [11]:

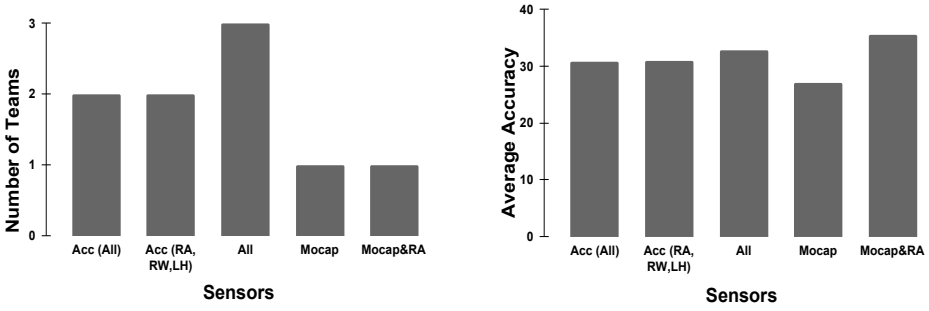
$$Accuracy(A_{mi}) = \frac{1}{n}\sum_{n}^{i=1}\frac{|Y_i \bigcap Z_i|}{|Y_i \bigcup Z_i|} \tag{2}$$

Here, Y is predicted labels and Z is actual labels of micro activities and n is Total number of samples. Then the average of two accuracies: macro activities and micro activities are calculated. The formula is as follows:

$$Accuracy,A = \frac{A_{ma} + A_{mi}}{2} \tag{3}$$

3.2 Results

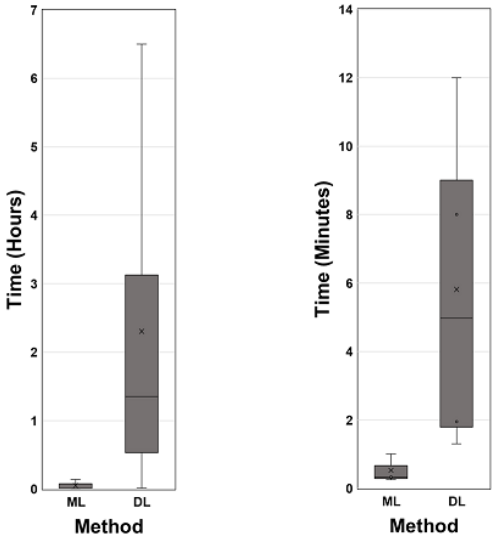
Participant teams used various pre-processing and classification methods for this *challenge*. Difference in the use of sensor modalities are also observed and it is shown in Figure 3 (a). As we can see in the figure, three teams used all of the sensors. There were four accelerometer sensors: right arm (RA), right wrist (RW), left wrist (LW) and left hip (LH). Different teams used different combinations of sensors as shown in the figure. Difference in accuracy by using different sensors can be seen in Figure 3 (b). Big difference in performance can be seen when **Mocap** and **Mocap&RA** is used. Best performance is achieved using the combination of Mocap and Accelerometer sensor in right arm (**Mocap&RA**). One possible reason can be, right arm is frequently used for cooking and all of the experiment participants were right handed person.



(a) (b)

Fig. 3: (a) Sensor modalities used by teams and (b) Accuracies by different sensor modalities

Training time and testing time of the teams is presented in Figure 4. For train and test, time was divided in Machine Learning (ML) and Deep Learning (DL) groups. Three teams used ML, five teams used DL and one team used both ML and DL. One of the team mentioned that it took a very long time for training and testing. So, only 8 teams' information is shown in the figure. Here, we can see that for training ML took very less time whereas DL took a long time. And for testing both of the groups required very less time. As for training it took longer, so it is shown in hours and testing took less than an hour, so to emphasize the difference between ML and DL, it is shown in minutes.

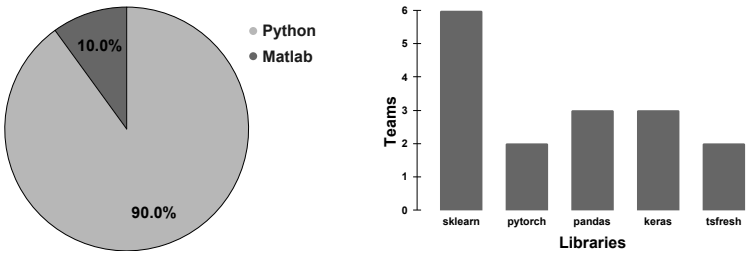


(a) (b)

Fig. 4: (a) Training time and (b) Testing time by classical machine learning (ML) and deep learning (DL) pipelines

As for the implementation, all of the teams used Python and one team used both Python and Matlab. This indicates the popularity of Python for AI and Machine learning related fields. Also in the Figure 5 (b) commonly used libraries are shown. The libraries which are used by multiple teams are shown in the figure.

For the window size, different teams used different window sizes. The dataset was provided in a 30 seconds window frame. Some of the teams modified it and 0.5, 2, 3, 10 seconds are used. Many teams used different feature extraction and selection strategies. Mean, standard deviation, max and min are commonly used features among different teams. One of the teams converted the inputs to image and used it for classification. For the post-processing of the data,use of resampling techniques and imputation strategies are observed. ML and DL pipelines used by



(a) (b)

Fig. 5: (a) Programming Language and (b) Libraries used by the teams

different teams are shown in Figure 6. It is visible that more people are using DL algorithms than ML, although ML is faster as seen in Figure 4.

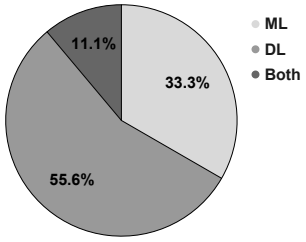
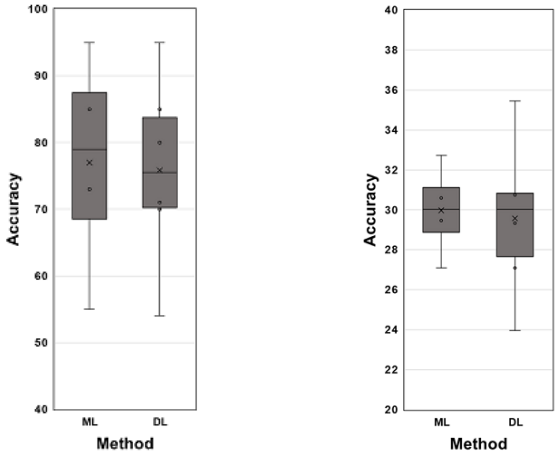


Fig. 6: ML and DL pipelines used by teams

Use of different ML algorithms like: *k*-NN, SVM, Random Forest and LightGBM was noted. In case of DL algorithms, use of CNN, GCN, Convolutional Neural Network (SCAR-Net), Deep Convolutional Bidirectional LSTM, Deep Convolutional GRU, ConvLSTM was observed. Accuracy comparison of these algorithms are done by dividing them into ML and DL groups, and it is shown in Figure 7 (a). This accuracy is only based on training data, provided by participants in their submitted papers. In the picture we can see that most of the DL algorithms have 70-85% accuracy during training while the range of accuracy for ML algorithms is quite large. One thing to notice that, the number of teams that used DL algorithms are almost double than the teams used ML, still having small range means that the performance of the DL algorithms are consistent and similar.



(a) (b)

Fig. 7: Accuracy of ML and DL pipelines for (a) Training Data and (b) Testing Data

On the other-hand, accuracy on the testing set is shown in Figure 7 (b).Samples of **subject 4** are used as testing data and no cross validation is used. Unlike training accuracy, the median of testing accuracy is smaller for ML algorithms than DL algorithms although the difference is small.

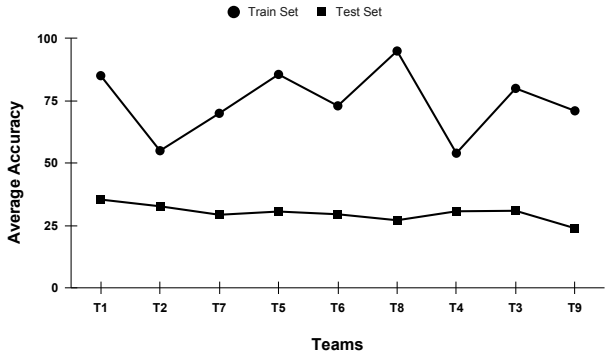


Fig. 8: Accuracy comparison of Training and Testing data

In Figure 8, comparison of accuracy of training and testing data is shown. Training accuracy is reported by participants in the papers and Testing accuracy is calculated by us using Equation 3. We can see a big gap in accuracy. All of the teams

performed very well, using training dataset, with lowest accuracy of 55%. But on testing data, the highest accuracy was 35.4%.

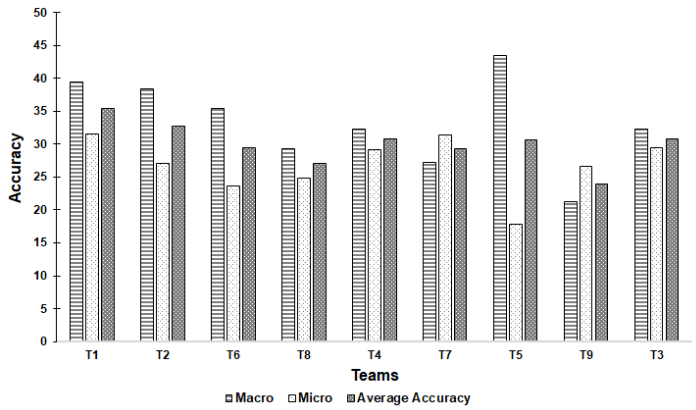


Fig. 9: Accuracy of Macro, Micro and All activities

Overall accuracy comparison on the test dataset for all the teams is shown in Figure 9. Here, the result is grouped in 3 groups: Macro, Micro and All. All means the average of Macro and Micro. The details of each is is given in Table 2. Overall, the recognition of micro activity was poorer than macro activity. One reason can be very few samples. Surprisingly 2 teams were able to recognize micro activity better, than macro activity. Team **T5** [16] has high accuracy for Macro (about 43%), but due to very poor performance of Micro activity classifier (17%), average accuracy degrades. For micro activity classification, team **T1** [18] and **T7** [15] does better (around 31%). On average the team **T1** performs better than all the other teams. Also it was observed that the difference in accuracy of the different teams are not much high.

		Truth		
		Cereal	**Fruitsalad**	**Sandwich**
Prediction	**Cereal**	**7**	**4**	**8**
	Fruitsalad	**10**	**15**	**10**
	Sandwich	**9**	**19**	**17**

Fig. 10: Confusion Matrix for Macro activities

Confusion matrix for the team with highest accuracy is shown in Figure 10. It can be seen that FRUITSALAD and SANDWICH have more samples correctly predicted than CEREAL. FRUITSALAD class has maximum number of samples incorrectly predicted, which had least number of samples during training. So we can assume that class imbalance has an effect on the detection of the labels. Similar scenario was seen for micro activities also.

4 Conclusion

After summarizing the results of this *challenge* we can say that, the average accuracy obtained by various teams ranges between 23% to 35%. Although for macro activity recognition, most of the team performed good, but the average results degraded because of comparatively lower performance of micro activity classification. The highest accuracy for macro activity classification is 43% whereas for micro it is 31%. So, we can conclude that detecting micro activity better, can result in higher average accuracy. Some classes like ADD, MIX- have very few samples. This can be one reason behind performance degradation of micro activity classification. Another case observed from the results is: although deep learning methods are very promising and showing excellent performance in many fields, but machine learning algorithms seem to perform better for this dataset. Also, the running time for the model is exceptionally lower for machine learning algorithms. One difference from previous *Nurse care activity recognition challenge* [21] is that, winning team of that *challenge* has lower training accuracy than test and this *challenge* has the opposite. One reason can be, the previous *challenge* dataset has larger number of sample than this *challenge*. So, in the future we want to collect data in wider range and share it with young researchers for more interesting results and observations.

Appendix

Table 2: Details of Each Team

Team	Cal. Accuracy	Used sensor modalities	Prog. language	Libraries	Classifier	Window size	Training time	Testing time	CPU	RAM	GPU
T1 (Chapter 2 [18])	35.45%	Mocap & RA	Python	Pytorch	CNN and GCN	NA	2 hours	12 min	12 Intel(R) Core(TM) i7-8700K CPU @ 3.70GHz	30 GB	Four pieces of GPU:GeForce GTX 1080 Ti
T2 (Chapter 3 [13])	32.74%	All	Python	sklearn, pandas, lightgbm, optuna	LightGBM and and Naïve Bayes	10 s	530 s	20 s	4.00GHz	32GB	No
T3 (Chapter 4 [14])	30.87%	Acc (RW, RA, LH)	Python, Matlab	Keras, Sklearn, Tensorflow	Deep Convolutional Bidirectional LSTM	10 s	6.5 hours	8 min	3.4GHz AMD Ryzen Threadripper 1950X 16-core	128 GB	2x 12 GB Nvidia GeForce GTX 1080 Ti
T4 (Chapter 5 [19])	30.75%	Acc (All)	Python	tensorflow, tsfresh, matplotlib	Convolutional Neural Network (SCAR-Net)	NA	4.2 minutes	1.3 min	Core i7	16 GB	RTX 2060
T5 (Chapter 6 [16])	30.59%	All	Python	keras, sklearn, sktime, pandas, tslearn, numpy	Multi-sampling classifiers (MSC)	30 s	1h 49min 12s		Intel Xeon E5-1620v3	DDR4 64 GB	GeForce GTX 1080
T6 (Chapter 7 [20])	29.47%	All	Python	pandas, sklearn, numpy	k-NN	30 s	15 s (train test together)		2.8 GHz Intel Core i7	16 GB	No
T7 (Chapter 8 [15])	29.33%	Acc (All)	Python	pytorch	ConvLSTM	500 ms	50.445 s	117.341 s	Intel Core i7-8700K 3.7KHz	DDR4 64GB	NVIDIA GeForce RTX 2080Ti GDDR6 11GB.
T8 (Chapter 9 [12])	27.09%	Mocap	Python	sklearn, xgb, hmm, tsfresh	Combination of different classifiers	2 s	<1 min	<1 min	Intel i7-4790	16GB	No
T9 (Chapter 10 [17])	23.95%	Acc (RW, RA, LH)	Python	keras, sklearn	Deep Convolutional GRU	3 s	Longtime	Longtime	Core i3	8GB	No

References

1. Cooking Activity Recognition Challenge, https://abc-research.github.io/cook2020/
2. 2nd International Conference on Activity and Behavior Computing (ABC2020), https://abc-research.github.io/
3. Stefano A, Marco A, Francesco B, Guglielmo C, Paolo C, and Alessio V (2012) A smartphone-based fall detection system. Special Issue on Pervasive Healthcare, Pervasive and Mobile Computing
4. Autism spectrum disorder (ASD), https://www.nimh.nih.gov/health/topics/autism-spectrum-disorders-asd/index.shtml
5. Dementia, https://www.who.int/news-room/fact-sheets/detail/dementia
6. Parkinson's disease (PD), https://www.parkinson.org/understanding-parkinsons/what-is-parkinsons
7. Paula L, Shingo T, Sayeda S A, Kohei A, Brahim B, Francois C and Sozo I (2020) A dataset for complex activity recognition with micro and macro activities in a cooking scenario (To appear)
8. Paula L, Shingo T, Kohei A, Sayeda S A, Moe M, Brahim B, Sozo I and Francois C (2020) Cooking Activity Dataset with Macro and Micro Activities, IEEE DataPort, doi: 10.21227/hyzg-9m49
9. Motion Capture Company, http://motionanalysis.com/movement-analysis/
10. Zhe C, Gines H, Tomas S, Shih-En W and Yaser S (2018) OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields, arXiv preprint arXiv:1812.08008
11. Mohammad S S (2010) A Literature Survey on Algorithms for Multi-label Learning. doi: 10.1.1.364.5612
12. Clément P, Vito J, Nina R, Martin G and Mitja L (2020) Identification of Cooking Preparation Using Motion Capture Data: A Submission to the Cooking Activity Recognition Challenge, Human Activity Recognition Challenge, Springer, 2020.
13. Ryoichi K, Roberto L, Kiyohito Y and Shinya W (2020) Let's not make it complicated - Using only LightGBM and Naive Bayes for macro and micro activity recognition from a small dataset, Human Activity Recognition Challenge, Springer, 2020.
14. Swapnil S S, Sandeep S S and Mani S (2020) Deep Convolutional Bidirectional LSTM for Complex Activity Recognition with Missing Data, Human Activity Recognition Challenge, Springer, 2020.
15. Atsuhiro F, Daiki K and Kazuya M (2020) Cooking Activity Recognition with Convolutional LSTM using Multi-label Loss Function and Majority Vote, Human Activity Recognition Challenge, Springer, 2020.
16. Ninnart F and Hakaru T (2020) Multi-Sampling Classifiers for the Cooking Activity Recognition Challenge, Human Activity Recognition Challenge, Springer, 2020.
17. Sadman S M, Omar S and Ahad M A R (2020) Cooking Activity Recognition with Varying Sampling Rates using Deep Convolutional GRU Framework, Human Activity Recognition Challenge, Springer, 2020.
18. Mao D, Lin X, Liu Y, Xu M, Wang G, Chen J and Zhang W (2020) Activity Recognition from Skeleton and Acceleration Data Using CNN and GCN, Human Activity Recognition Challenge, Springer, 2020.
19. Zabir A N (2020) SCAR-Net: Scalable ConvNet for Activity Recognition with multi-modal Sensor Data, Human Activity Recognition Challenge, Springer, 2020.
20. Shkurta G, Elena D L and Silvia S (2020) Multi-class Multi-label Classification for Cooking Activity Recognition, Human Activity Recognition Challenge, Springer, 2020.
21. Paula L, Sayeda S A, Shingo T, Tittaya M, Nattaya M, Farina F, Yusuke N, Kohei A, Tsuyoshi O, François C and Sozo Inoue (2019) Nurse Care Activity Recognition Challenge: Summary and Results. In: UbiComp/ISWC '19 Adjunct: Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers