Quantifying Quality of Actions Using Wearable Sensor

Mohammad Al-Naser[1,2], Takehiro Niikura[3], Sheraz Ahmed[1], Hiroki Ohashi[4], Takuto Sato[4], Mitsuhiro Okada[4], Katsuyuki Nakamura[4], and Andreas Dengel[1,2]

[1] German Research Center for Artificial Intelligence (DFKI), Kaiseslautern, Germany
[2] TU Kaiserslautern, Kaiserslautern, Germany
{mohammad.al_naser, sheraz.ahmed, andreas.dengel}@dfki.de
[3] Hitachi Europe, Kaiserslautern, Germany
takehiro.niikura@hitachi-eu.com
[4] Hitachi Ltd., Tokoy, Japan
{hiroki.ohashi.uo, takuto.sato.hn, mitsuhiro.okada.uf, katsuyuki.nakamura.xv}@hitachi.de

Abstract. This paper introduces a novel approach to quantify the quality of human actions. The presented approach uses expert action data to define the space in order to gauge the performance of any user to identify expertise level. The proposed approach uses pose estimation model to identify different body attributes (legs, shoulders, head ....) status (left, right, bend, curl ...), which is further passed to autoencoder to have a latent representation encoding all the relevant information. This encoded representation is further passed to OneClass SVM to estimate the boundaries based on latent representation of expert data. These learned boundaries are used to gauge the quality of any questioned user with respect to the selected expert. The proposed approach enables identifying any critical situations in real work environment to avoid risky positions.

Keywords: Autoencoder · OneClass SVM · Actions evaluation · Wearable sensor.

1 Introduction

Human-activity recognition (HAR) is one of the important research topics for the systems involving human-machine system [9, 13]. It has a wide variety of applications such as health care, rehabilitation, education, sport, and worker assistance [25, 20, 17, 15, 19]. Especially, if the system can assess the quality of action and gives moderate feedback to the users about how they can improve. Forevermore, it can be used for training as well as to avoid any critical situations.

With the recent progress in deep neural networks (DNN), the performance of HAR system has been significantly improved both in terms of accuracy and the number of actions [7, 10, 22, 24]. While state-of-the-art HAR [22, 7] research accomplished promising performance for many action recognition tasks, most of

the researches mainly focused on recognizing which action is being performed [6, 10, 22]. Though these researches are meaningful and can be utilized for an application to recognize whether the sequence of action executed or not, they do not focus on identifying the quality of recognized action. We found only limited number of research on how to assess the quality of action [7, 8, 15] and most of them based on video sequence analysis.

In this paper, we propose a novel approach to quantify the quality of action using wearable sensor, in which we use autoencoder and OneClass SVM. In contrast to the existing approaches [7, 7, 18] which work only with video data; the proposed approach deals with sensors data that obtained from wearable sensors (namely perception neuron [2]) to assess the quality of various actions. Sensors in the wearable device directly measure the movements of more than 30 body parts, and collected data is used to create attributes of the body joints, the attributes passed to the autoencoder to create the latent space. Then the model learns to estimate the boundaries based on latent representation of expert data. These learned boundaries are used to assess the quality of any questioned user with respect to the correspondent expert. The proposed approach provide both coarse as well as fine grained information about quality of an action. This means, it firstly provides an overall information about how good a particular action is. Then for fine details, it highlights part of the action which can be improved to increase overall quality of that particular action.

Our main contributions are as follows:

- We propose a novel method to assess the quality of various actions sensed by wearable sensor.
- We provided a new human action dataset using wearable sensors to be used for actions assessment.

The experimental results showed that our model achieved high accuracy for most of the actions.

2 Related Work

Only a limited number of research tackled the problem of action quality assessment [18, 20, 23]. Lv et al. proposed a system [14] which assesses the quality of driving behaviors based on radio signals. They used hand-crafted features for quality assessment, and the system can be used for differentiating a triple body status and for identifying among 15 drivers with high accuracy. An efficient system to detect and classify several swing motions in various kinds of sports has been developed by Anand et al. [3], by utilizing IMU sensor on users wrist. Regarding the movement assessment for sport, there are some other studies [12, 15], for example, Bacic [4] used vision-based motion capture system to analyze the swing motion of tennis whether the swing is good or bad. Although some of the research above aimed to develop a general system to assess the movements of users, their systems rely on some features and methods which are specific to their target movement, implicitly or explicitly.

Velloso et al. [21] also presented a system which assesses the performance of users in real time and provides feedback on how to improve their performance. Their system is also designed to their specific target action. However, they gave some important indications in their paper. They said, "This evidences that specifying a movement by natural language and estimating precise angles by observation is difficult", and therefore they adopted Programming by Demonstration (PbD) approach. PbD aims to make it possible to program systems by having a user demonstrate to them how they should behave, instead of hardcoding a systems behavior. This is a very important indication for us, because it is often the case that experts cant explain the key point of their skill by natural language. In other words, they have muscle memories about how they should execute, but it is very difficult for them to verbalize their skill. In this sense, PbD approach seems promising for our purpose.

In the domain of image processing research, deep-ranking approach has recently achieved a great success in skill assessment. Doughty et al. [7] presented a general method for assessing skill from video. The authors collected egocentric videos from both experts and novices, and defined which video has better skill level than the other. With this information, the model learned how to assess skill, like surgery, drawing, and rolling pizza dough, and so on. One advantage of their approach is that their model can deal with many kinds of activities, since the model is designed independently from task specific information. This is important because it can assess many kinds of actions with such generalization capability. Also, Doughty et al. [8] presented a temporal attention modules to determine relative skill from long videos, they use a rank-aware loss function to train a temporal attention model. The model learns to attend task-relevant video parts. Also they proposed a joint loss trains two attention modules to separately attend to video parts, which are indicative of higher (pros) and lower (cons) skill. Although the models show a very good results in skill determination, but such settings is difficult to use in the working environment, because of the privacy issues and the occlusions that usually happens for the video.

3 Target Actions and Data

3.1 Target Actions

Since most of the existing methods are based on video analysis, there is no publicly available dataset for evaluating the quality of action, we need to construct a dataset by ourselves.

Supposing that actions are in a real scenario, the key point of skills can exist at any part of the body. For example, if a user is using screw driver, the movement of arms and/or hands might be important, however if the user needs to use it in a narrow space, the movement of legs and waist are also important. Therefore, we designed our experiments where target actions are defined based on the following criteria.

– As a whole, target actions contain movements of whole body.

– To confirm the validity of our system after the development, both good action and bad action need to be definable by an existing metric.

Table 1. Target actions definition

action	**pick up**	**hold**	**move**
Good	Bend knees Put body closer to Object	Stretch elbows Keep object closer to Body	Use legs Don't twist waist
Bad	Don't bend knees Bend waist Keep body distant from object	Bend elbows Keep object distant from body	Twist waist

Based on these requirements, we chose three actions, which are 'pick up', 'hold', and 'move' an object 1. These three actions compose an activity of carrying an object, and to carry an object the worker needs to use his/her whole body. Also, it's important that these actions are basic and common in any real scenario. Additionally, there's a global metric for estimating workload with respect to workers posture, named as OWAS (The Ovako Working posture Assessment System). By using this metric, we can define both good action and bad action. In this paper we call the good action (expert) and bad action (novice), where we think the good skills lead to less workload. Figure 1 shows images of good

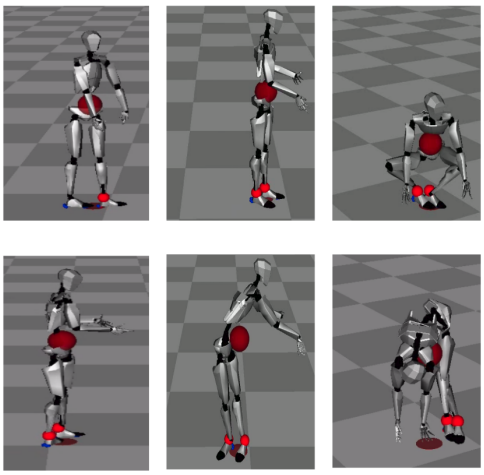

Fig. 1. Target actions. The top images are the good actions(expert) and the lower ones are the bad actions(novice). Where the left column is 'hold' action, the middle is 'move' action, and the right is 'pick up' action.

and bad manner for each action, and Table 1 shows the definition of good and

bad manner for each action. In 'pick up' action, a user picks up and puts down a printer from/on the floor. The key point of this action is the movement of waist and knees, and a worker may hurt his/her back in bad action. In 'hold' action, a worker keeps holding a projector, and he can walk around freely while holding it, and the key point for this action is the usage of elbows and upper arms. The bad manner of this action easily causes fatigue on upper arms. In 'move' action, a worker moves a projector from front to left/right side. In this case, if the worker twists his/her waist while 'move' action, it may damage his/her waist.

3.2 Wearable Sensor

Our goal is to quantify the quality of full-body action. Thus, a very dense sensor set across full-body is required. Perception Neuron from Noitom Ltd [2] as shown in figure 2 is one of the best commercial products that satisfies this requirement and it is available in the market. It has 31 IMU sensors across full body; 1 on head, 2 on shoulders, 2 on upper arms, 2 on lower arms, 2 on hands, 14 on fingers, 1 on spine, 1 on hip, 2 on upper legs, 2 on lower legs, 2 on feet. Each IMU is composed of a 3-axis accelerometer, 3-axis gyroscope and 3-axis magnetometer.

Fig. 2. The perception neuron sensor.

3.3 Dataset

We collected data of three actions in the laboratory. Table 2 gives the conditions when we collected the data. As we mentioned before, the target actions are 'pick up', 'hold', and 'move'. 11 participants joined the data collection. We collected data 3 times for both good action and bad action respectively, and 1 trial of data collection lasted for 1 minute. During 1 trial, participants were asked to perform the same action repeatedly. The definition of one repetition for 'pick up' is to pick up a printer and to put it down on the floor, and for 'move' is to move a projector from front to right, turn to the printer, and move it to the original position again. We collected data from 29 IMU sensors, all sensors without on feet, across full body. Since the magnetometer provides the position of the sensor in quaternion, each IMU provides 10 types of data at 60 fps.

Table 2. Conditions of collecting action data

Number of actions	3 (Pick up, Hold, Move)
Number of participants	11
Number of trials for each action	Total: 6 (3 for good action, 3 for bad action)
Duration time for 1 trial	1 minute

4 Proposed Method

Our model is inspired by anomaly detection methods [5][11] which have a very good performance in outlier detection. These methods use the reconstruction loss of the autoencoder to detect the anomaly data, where the normal data have lower reconstruction loss than the outlier data as shown in figure 4. The model use only good actions (expert) data to define the expert space, to compare the

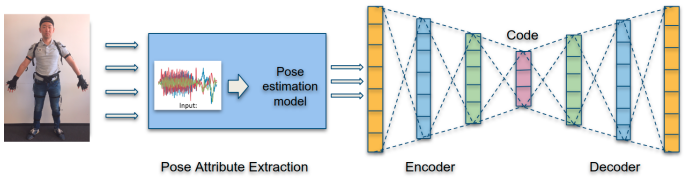

Fig. 3. Overview of the initial model.

performance of any user in order to identify the action level.

Table 3. Pose estimation attributes which are used as an input for the autoencoder. It's to be noted that some of them are binary attributes, while others have continues values

Joint	**Type**	**Value**
head	classification	up, down, left, right, front
shoulder	classification	up, down, left, right, front
elbow	regression	0 (straight)-1 (bend)
wrist	regression	0 (reverse curl)-1 (curl)
hand	classification	normal, grasp, pointing
waist	classification	straight, bend, twist-L, twist-R
hip joint	regression	0 (straight)-1 (bend)
knee	regression	0 (straight)-1 (bend)

In our model we employed a zero-shot pose estimation model [16]. The sensors raw data passed to the model to estimate the status of 14 major human-body joints, which we call attributes. The attributes represent various body poses,

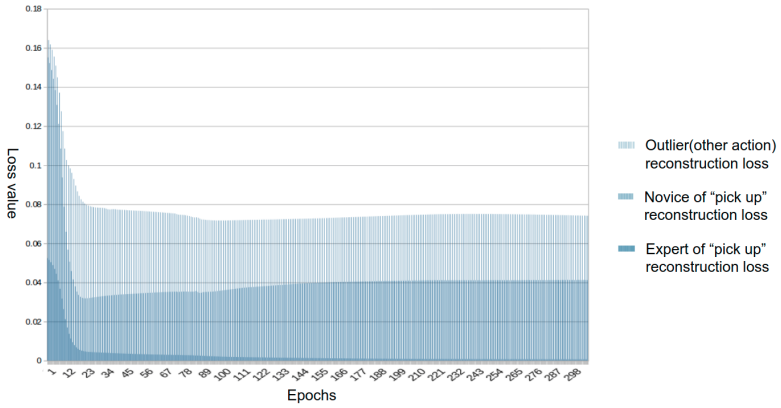

Fig. 4. Reconstruction loss values for 'pick up' action.

Where multi-class classification or regression are used for the joints depending on their characteristics, see table 3. Note that each joint has left part and right part except head and waist. In the next step, we passed the attributes to the autoencoder to create the latent space of the good skills. The autoencoder which we use is deep autoencoder consists of three fully connected layers, and the reconstruction loss is mean square error. The overview of deep autoencoder is shown in Figure 3. After testing the model with good actions (expert) and bad actions (novice), we found the reconstruction loss of the expert less than the novice reconstruction loss, as shown in Figure 4, which means the autoencoder succeeded in learning the differences between the good and bad actions.

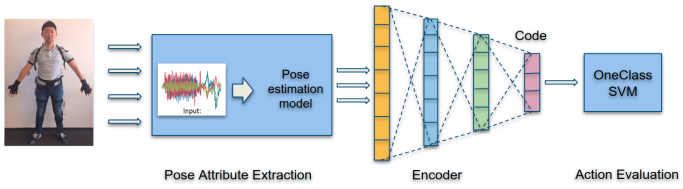

Fig. 5. Overview of encoder and OneClass SVM.

Then, we developed the model to score the actions. We substituted the decoder part in the autoencoder by OneClass SVM model, as shown in Figure 5. For OneClass SVM, we used RBF(radial basis function) kernel since we are dealing with non-linear problem. The OneClass SVM learns the latent space of the experts skills, so that it can output the decision scores on the test data.

5 Training and Evaluation

5.1 Training

For training, evaluation, and testing, we manually separated the recorded data of 1 trial into several splits, which contains one repetition of action. We created 6 splits for 'Pick up' and 5 splits for "Move". As we recorded 'hold' action continuously, we don't have clear cut to create splits. For this experiment we divided the whole action data into 6 same duration splits. Table 3 shows the number of splits for each action.

Then, we separated the data into training, validation and test data. Since there is a lot of individual variations in data, we separated data in terms of person so that our model will be tested with completely new data in validation and test phase.

We train our model by feeding the sensors data to the pose estimation model, the model use 0.5 second time window to estimate the attributes of the body (joints status), where the number of the estimated attributes is 33 for each window.

Table 4. Number of splits for each action

Action	**Number of splits from 1 trial**	**Number of splits in total**
Pick up	6	396 (Expert: 198, Novice: 198)
Hold	6	396 (Expert: 198, Novice: 198)
Move	5	330 (Expert: 165, Novice: 165)

Then we concatenate 5 window attributes together to train the autoencoder as shown in figure 6. While training we save the latent space for each batch until the epoch is finished. Then we train the OneClass SVM by the saved latent space. We tried to train the OneClass SVM by latent space of each batch separately, but the results was worse than the training by the latent space for the whole data. Next step is to validate the OneClass SVM model using expert and novice data, and we keep doing this to the last epoch.

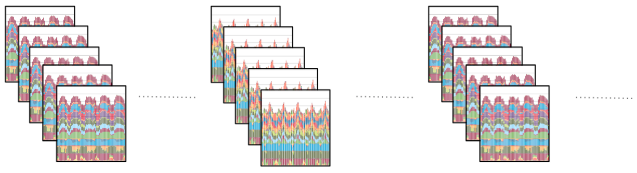

Fig. 6. Each 5 window attributes concatenated together then trained on the autoencoder

Also, we want to check how the increase of the training data affects the results, So we gradually increased the number of people for training data from 2 to 5, one by one. Practically, we always regarded one specific person as a test data. At the beginning of experiment, we used only the data from 2 participants as training data, and did the rest as validation data. After each experiment, we took 1 participant from validation data and added it in training data.

5.2 Evaluation

Finally, after training our model, we evaluated the model with accuracy, which is calculated with the following equation.

$$accuracy = \frac{L_{correct}}{L_{total}} \tag{1}$$

Where, L_{total} is the number of splits for test data. And $L_{correct}$ is the number of splits in which the score for expert was higher than 0, and the score is less than 0 for the novice.

6 Results

As described in the training section, we trained the model with 2 participants then increased the training data one by one to 5, we stopped at 5 participants because we didn't find improvement in performance with the increase of the training data. Where the model has already showed a very good performance with 2 training participant, as shown in table 5. So all the presented results are for the model trained by 2 participants. And note that the avatar in figures 7 and 8 is for the participant during the experiment, it's created by the AXIS Neuron the perception neuron software [1]. We can see example of the scores for the target activities in figure 7, we can notice that the model has a very good performance in evaluating these actions. Note that the decision scores of OneClass SVM becomes positive when the data is recognized as an expert, and goes negative when recognized as a novice.

Table 5. The accuracy of the model for each action

	Pick up	**Hold**	**Move**
Expert	98%	99%	76.4%
Novice	99%	99%	76%

Also, to validate our results and to know which epoch model is the best to use, we tested in each epoch one expert data and one novice data and plotted the results as it appears in figure 10. This figure shows the average score of experts and novices over the epochs. It is clear that the model was able to learn and score the actions after a few epochs. We can notice from the results as shown

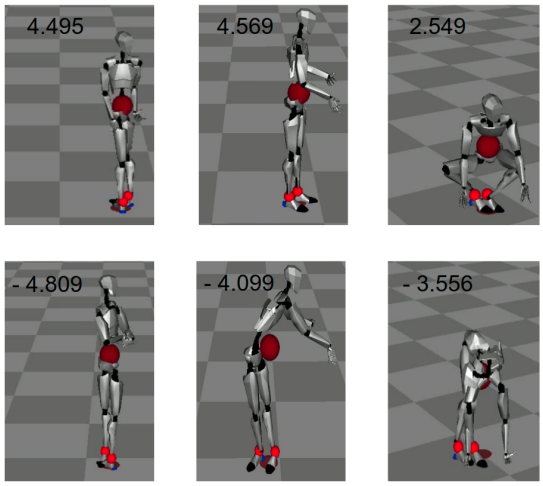

Fig. 7. Example of results for all actions. Top images for good actions, bottom images bad actions, where left is 'hold', middle 'move', and right 'pick up' action.

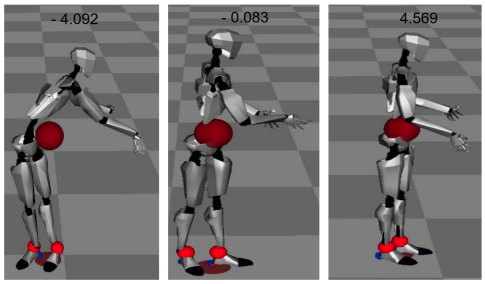

Fig. 8. Three examples of the 'move' action. We can see that with more twist in the waist the more negative the score. From left the novice to the right the expert.

in table 5 that the most challenging action for the model is 'move'. We assume it's because of the waist, which is the key attribute to identify novice and expert for this action, the waist has no clear joint movement as the other joints in the body. Still the model proved that it can evaluate the skills of the participants comparing to the expert action. Figure 8 and 9 show examples of results for experiment participants of 'move' and 'pick up' actions. We can see that when the performed action is closer to the good action (expert), the score becomes higher (4.56), and it gets gradually lower when the performance get closer to the bad action (novice).

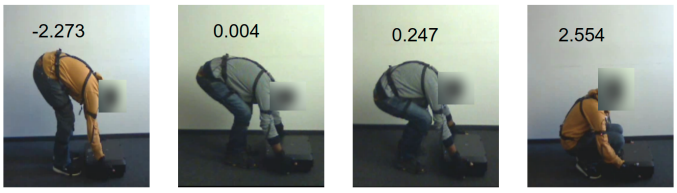

Fig. 9. Four examples of the 'pick up' action. We can notice that with more bend in the knees and less bend in the back the more positive the score is. Left the most novice to the right the most expert

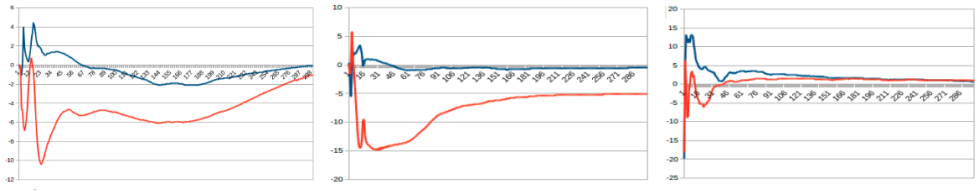

Fig. 10. Average score of experts (blue) and novices (red) over epochs. Where the left is 'Hold' action, middle for 'Pick up', and right for 'move' action. Where x-axis is the epochs and y-axis is the score

7 Discussion

A quantification of action model has been developed to assess actions in real scenarios. Our initial model consists of body pose estimation model followed by autoencoder. Then we developed this model to score the action level. As shown in the results section, the recognition accuracies of quality evaluation is around 99% for 'pick up' and 'hold' action, and 76% for the 'move' action.

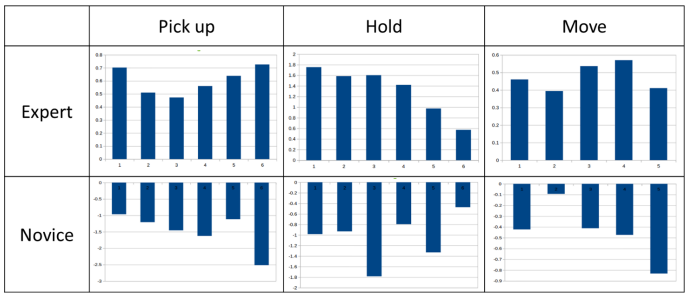

Fig. 11. OneClass SVM scores (each column represents one split)

Figure 10 shows average scores of experts and novices over epochs. We notice that the model can learn the differences between expert and novice in all three

actions, still the difference in the score value for the 'move' action is close between expert and novice, also we can see this in Figure 11. To analyze further we checked the autoencoder reconstruction loss for this action, see figure 12. It's clear that the novice reconstruction values are close to the expert values, and its getting closer over epochs, which is not the case for the other actions, see figure 4.

Then, why our model have difficulties only for 'move' action? We assume mainly 2 reasons to this problem. The first reason is that 'move' action has only small differences between expert and novice action, compared to other 2 actions. The second reason is that the sensor we used may not provide enough data to

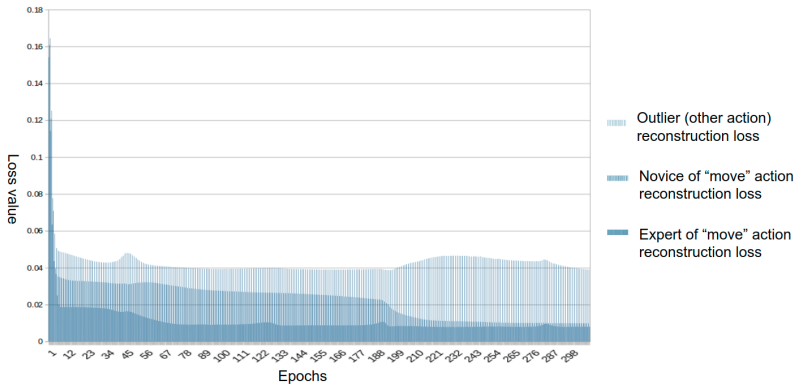

Fig. 12. Reconstruction loss for 'move' action. The values of the novice (bad) movement are close to the expert (good) values.

recognize twisting waist. Twisting waist can be explained as the direction of upper body rotates to left or right while the direction of lower body remains as it is. Perception Neuron has 1 IMU sensor on spine, and it can work to detect the rotation of upper body, however it has no IMU sensor which is suitable to detect the direction of lower body. We think by improving the waist status representation the model will perform better as the 'pick up' and 'hold' action.

8 Conclusion

We presented a novel action quality quantifying model uses only the expert data to learn the good skills. The base of our architecture is pose estimation model followed by autoencoder, then on top of it a OneClass SVM model. The model learn to estimate the boundaries based on latent representation of expert data. These learned boundaries are used to assess the quality of any questioned user with respect to the selected expert. The model showed that it can asses the actions quality in high performance, where it achieved accuracy of 99% for 'hold' and 'pick up' actions, and 76% for 'move' action.

References

1. AXIS Neuron, https://neuronmocap.com/content/axis-neuron-software
2. Perception Neuron, https://www.noitom.com/solutions/perception-neuron
3. Anand, A., Sharma, M., Srivastava, R., Kaligounder, L., Prakash, D.: Wearable motion sensor based analysis of swing sports. In: 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA). pp. 261–267 (Dec 2017). https://doi.org/10.1109/ICMLA.2017.0-149
4. Bavci'c, B.: Towards the next generation of exergames: Flexible and personalised assessment-based identification of tennis swings (2018)
5. Borghesi, A., Bartolini, A., Lombardi, M., Milano, M., Benini, L.: Anomaly detection using autoencoders in high performance computing systems. CoRR **abs/1811.05269** (2018)
6. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp. 4724–4733 (2017)
7. Doughty, H., Damen, D., Mayol-Cuevas, W.W.: Who's better? who's best? pairwise deep ranking for skill determination. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition pp. 6057–6066 (2018)
8. Doughty, H., Mayol-Cuevas, W.W., Damen, D.: The pros and cons: Rank-aware temporal attention for skill determination in long videos. CoRR **abs/1812.05538** (2018)
9. Gaglio, S., Re, G.L., Morana, M.: Human activity recognition process using 3-d posture data. IEEE Transactions on Human-Machine Systems **45**(5), 586–597 (2015)
10. Jordao, A., Jr., A.C.N., de Souza, J.S., Schwartz, W.R.: Human activity recognition based on wearable sensor data: A standardization of the state-of-the-art. CoRR **abs/1806.05226** (2018), http://arxiv.org/abs/1806.05226
11. Kiran, B.R., Thomas, D.M., Parakkal, R.: An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos. J. Imaging **4**, 36 (2018)
12. Ladha, C., Hammerla, N.Y., Olivier, P., Plötz, T.: Climbax: skill assessment for climbing enthusiasts. In: UbiComp (2013)
13. Li, M., Wei, J., Zheng, X., Bolton, M.L.: A formal machinelearning approach to generating humanmachine interfaces from task models. IEEE Transactions on Human-Machine Systems **47**(6), 822–833 (Dec 2017). https://doi.org/10.1109/THMS.2017.2700630
14. Lv, S., Lu, Y., Dong, M., Wang, X., Dou, Y., Zhuang, W.: Qualitative action recognition by wireless radio signals in humanmachine systems. IEEE Transactions on Human-Machine Systems **47**(6), 789–800 (Dec 2017). https://doi.org/10.1109/THMS.2017.2693242
15. Mller, A., Roalter, L., Diewald, S., Scherr, J., Kranz, M., Hammerla, N., Olivier, P., Pltz, T.: Gymskill: A personal trainer for physical exercises. In: 2012 IEEE International Conference on Pervasive Computing and Communications. pp. 213–220 (March 2012). https://doi.org/10.1109/PerCom.2012.6199869
16. Ohashi, H., Al-Naser, M., Ahmed, S., Nakamura, K., Sato, T., Dengel, A.: Attributes importance for zero-shot pose-classification based on wearable sensors. In: Sensors (2018)
17. Parisi, G.I., Magg, S., Wermter, S.: Human motion assessment in real time using recurrent self-organization. 2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN) pp. 71–76 (2016)

18. Parmar, P., Morris, B.T.: Learning to score olympic events (July 2017). https://doi.org/10.1109/CVPRW.2017.16
19. Parmar, P., Morris, B.T.: Learning to score olympic events. CoRR **abs/1611.05125** (2016), http://arxiv.org/abs/1611.05125
20. Pirsiavash, H., Vondrick, C., Torralba, A.: Assessing the quality of actions. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) Computer Vision – ECCV 2014. pp. 556–571. Springer International Publishing, Cham (2014)
21. Velloso, E., Bulling, A., Gellersen, H.: Motionma: Motion modelling and analysis by demonstration. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 1309–1318. CHI '13, ACM, New York, NY, USA (2013). https://doi.org/10.1145/2470654.2466171, http://doi.acm.org/10.1145/2470654.2466171
22. Wang, J., Chen, Y., Hao, S., Peng, X., Hu, L.: Deep learning for sensor-based activity recognition: A survey. Pattern Recognition Letters **119**, 3–11 (2018)
23. Wnuk, K., Soatto, S.: Analyzing diving: A dataset for judging action quality. In: ACCV Workshops (2010)
24. Zhang, W., Qin, L., Zhong, W., Guo, X., Wang, G.: Framework of sequence chunking for human activity recognition using wearables. In: Proceedings of the 2019 International Conference on Image, Video and Signal Processing. pp. 93–98. IVSP 2019, ACM, New York, NY, USA (2019). https://doi.org/10.1145/3317640.3317647, http://doi.acm.org/10.1145/3317640.3317647
25. Zia, A., Sharma, Y., Bettadapura, V., Sarin, E.L., Clements, M.A., Essa, I.: Automated assessment of surgical skills using frequency analysis. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A. (eds.) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. pp. 430–438. Springer International Publishing, Cham (2015)