Activity Recognition in Industrial Environment using Two Layers Learning

Robin Fays, Engineering, Haute Ecole de Namur-Liege-Luxembourg-Pierrard, Belgique, robin.fays@henallux.be (mailto:robin.fays@henallux.be)
Rim Slama, Engineering, Haute Ecole de Namur-Liege-Luxembourg-Pierrard, Belgique, rimslamarim@gmail.com (mailto:rimslamarim@gmail.com)

DOI: https://doi.org/10.1145/3503961.3503966 (https://doi.org/10.1145/3503961.3503966)
VSIP 2021: 2021 3rd International Conference on Video, Signal and Image Processing (https://doi.org/10.1145/3503961), Wuhan, China, November 2021

Action and activity recognition is essential in the world of cobots to ensure the best efficiency and a safety collaboration between a robot and the human-being. The approach of the article is the creation of a new activity dataset for an industrial context with cobots for recognition. We proposed to use LSTM (Long Short-Term Memory) to analyse and recognize the activities and we also proposed to model the action using the Principal Component Analysis (PCA) and then recognize the activity using LSTM. Using this two level approaches on the dataset we collected, we obtained high recognition level : 96.826 (+/-0.383) %.

CCS Concepts: Computing methodologies, Artificial intelligence, Computer systems organization, Architecture, Other architectures, Neural networks, Networks, Network algorithms

KEYWORDS: Activity, Recognition, Deep Learning, Cobotics, Industrial, LSTM

ACM Reference Format:
Robin Fays and Rim Slama. 2021. Activity Recognition in Industrial Environment using Two Layers Learning. In *2021 3rd International Conference on Video, Signal and Image Processing (VSIP 2021), November 19-21, 2021, Wuhan, China.* ACM, New York, NY, USA, 9 Pages. https://doi.org/10.1145/3503961.3503966 (https://doi.org/10.1145/3503961.3503966)

1 INTRODUCTION

The use of industrial robots is constantly increasing inside companies. It can be used alone or for assistive tasks. The second use means it is a collaborative robots (cobot) made for working with humans as an assistant in a task or process (see figure 1) or as a guide. The understanding of the behavior and motions of humans is thus extremely essential and can be complex because of the parallelism between the performance of the cobot and the moment the cobot collects the human's information [1]. Furthermore, the safety of humans is mandatory in this context.

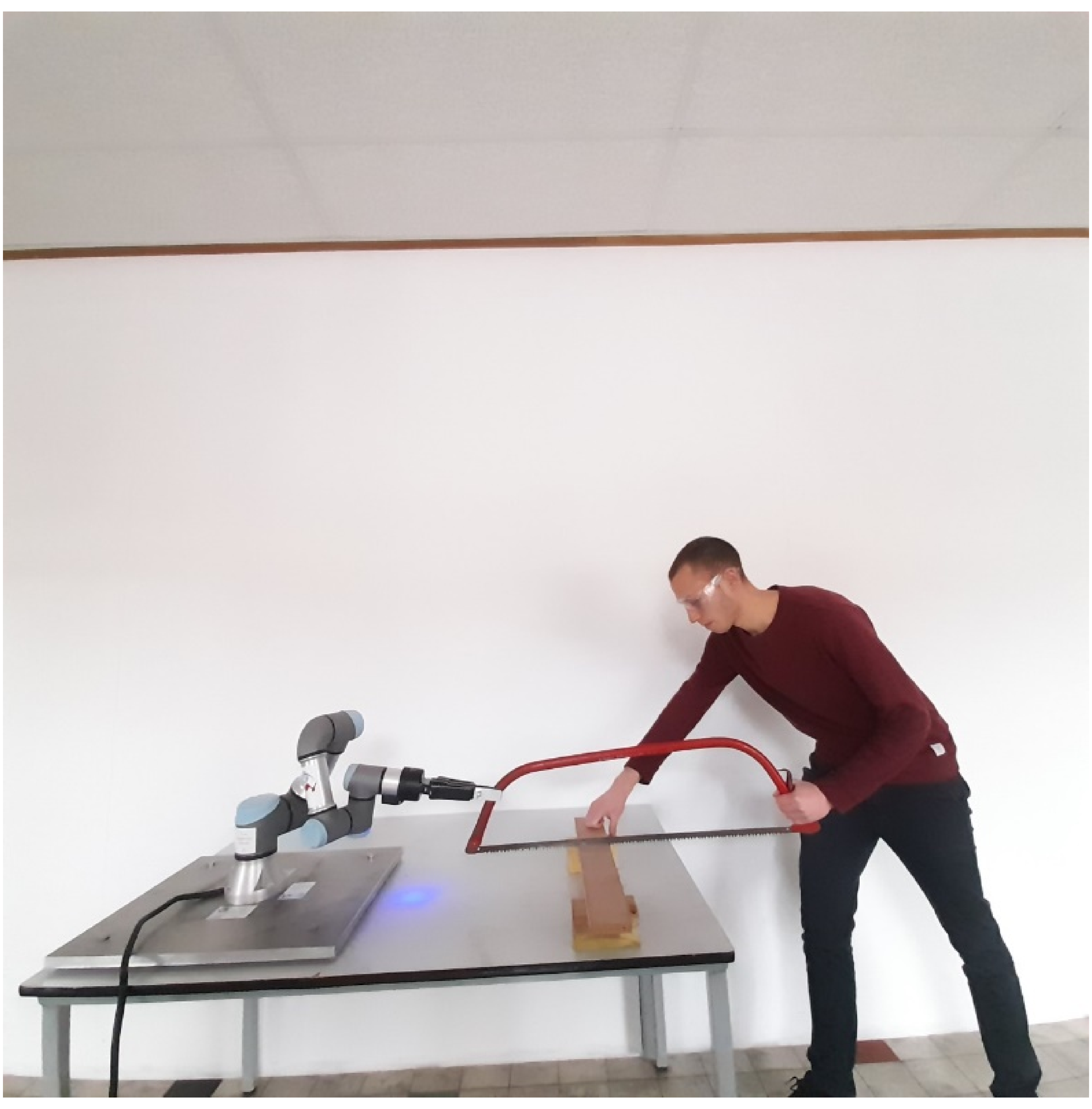

Figure 1: Example of human robot collaboration in the context of industrial work

Even if it is still difficult to perfectly understand the human motions and behavior, action and activity (combination of actions) recognition algorithms improved considerably due to the high evolution of depth cameras. The understanding of activities can be difficult because an activity is composed of several actions which are not always the same. For instance, the activity "starting to work" can include the action "take off jacket" if we are in winter but not if we are in summer.

In this work, we propose a combination of several actions from the dataset NTU RGB+D [2], for creating a new dataset composed of activities related to the industrial environment for human-robot interaction (HRI). The goal of this dataset is to help workers in a cobot application, especially in accuracy tasks such as drilling at the right place. The main contributions of this work are summarized as follows:

- We create a new industrial activities dataset from the actions of the NTU RGB+D dataset. We did not find any when the project started so that is why we have created ours.
- We evaluate this new dataset with a deep learning algorithm based on time series analysis and precisely LSTM algorithm.
- We evaluate the efficiency of the use of the Principal Component Analysis (PCA) as representation of the action and then feed the LSTM algorithm that we tuned to obtain interesting results in order to have a double-layer algorithm.

2 STATE-OF-THE-ART

Due to the evolution of the robots, the implementation of robots for interactive work with human-beings increases a lot. Some work has already be proposed [3,4] to study the collaborations. Song et al. [3] explores "the characteristics of the action recognition task in interaction scenarios and propose an attention-oriented multi-level network framework to meet the need for real-time interaction". Akkaladevi et al. [4] proposed a "multi-label human action recognition framework with the capability to detect multiple actions simultaneously in real-time". In these work, the importance of action recognition is necessary in order to provide the best efficiency and the best safety to the project.

During the past few years, action recognition received a lot of attention and a lot of methods have been proposed. A survey has been done in [5] and explored several possibilities of actions recognitions such as recognition based on images. For instance, Varol et al. [6] demonstrates the efficiency of the temporal extents in this field. Another method is action recognition based on joint skeleton such as [7] that proposed a method by capturing relevant local dynamics.

One of the algorithms used in action recognition is the Recurrent Neural Network (RNN) such as in [8] for learning the long-term dependencies between the features. The main disadvantage of this neural network is the vanishing gradient problem corresponding to the gradients of the neural network becoming too small if the length of the network is too high.

A variant of Recurrent Neural Network called Long Short-Term Memory (LSTM) [9] appeared for solving the vanishing gradient problem. The advantages of the LSTM are that it will avoid the long-term dependency problem of the RNN [10] and allows a better remembering. Chung et Al. [11] describes the efficiency of the LSTM related to the traditional RNN with the speech recognition.

Before using these algorithms, it is possible to modify the data in order to improve the recognition rate, it is the data pre-processing. For instance, Yan et al. [12] suggested a spatial temporal structure as input in the algorithm. Another method is the use of the Principal Component Analysis (PCA) as pre-processing before learning or as reduction of data [13] or a pre-processing by reducing the data followed such as in [14]. The goal of the PCA is to extract the important features of a bunch of data. It reduces the number of data and the time required for the neural network. The problem in the reduction of the number of data is that it can skip the most important information, so the loss of information can be detrimental to recognition. That is why we wanted to evaluate the efficiency of this pre-processing in this work. Data pre-processing and algorithms are used in action recognition but nowadays, activity recognition appeared.

An activity is an assembly of consecutive actions. Several works addressed the task of activity recognition with skeleton features. The Watch-n-Patch dataset [15] is an unsupervised activity dataset. Wu et al. provided a method studying the co-occurrence and temporal relations of actions in an activity [16].

3 OUR APPROACH

In this section, we introduce the proposed method. We first present our new dataset created by the assembly of several actions. The sequence of the stage-descriptor used which is the joints of the skeleton of each resampled frames. We then introduce the data pre-processing we used : principal component analysis. Finally, we present the algorithm used for testing the efficiency which is a LSTM because it provides good results, such as in [17].

3.1 Creation of the Database

In action recognition with the skeleton, each action is described by 3D-joints (x,y,z). The joints provided by the dataset are 25 3D-joints as shown in figure 3. In our, we only focus on the temporal stage, so the 3D joints of every consecutive frames. For each action sample, it is resampled to 80 steps (the average number of steps, the minimum is 0 and the maximum is 300) for having the same number of data and unifying the representation. The temporal stage for each action is thus represented as:

$$x_p = \left[(x_1)^1, (y_1)^1, (z_1)^1, (x_2)^1, \ldots, (x_{25})^{80}, (y_{25})^{80}, (z_{25})^{80}\right] \tag{1}$$

We create 10 labelled activities from the database NTU RGB+D [2]. For each activities, we concatenate 3 individual actions acted by the same performer (P), captured by the same camera (C) during the same set up number (S) and the same replication (R), see figure 2.

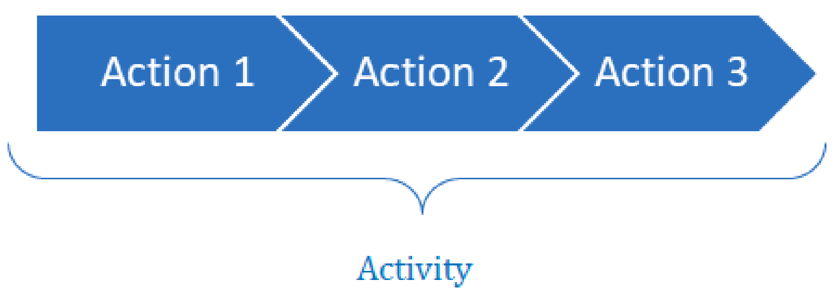

Figure 2: Creation of activities

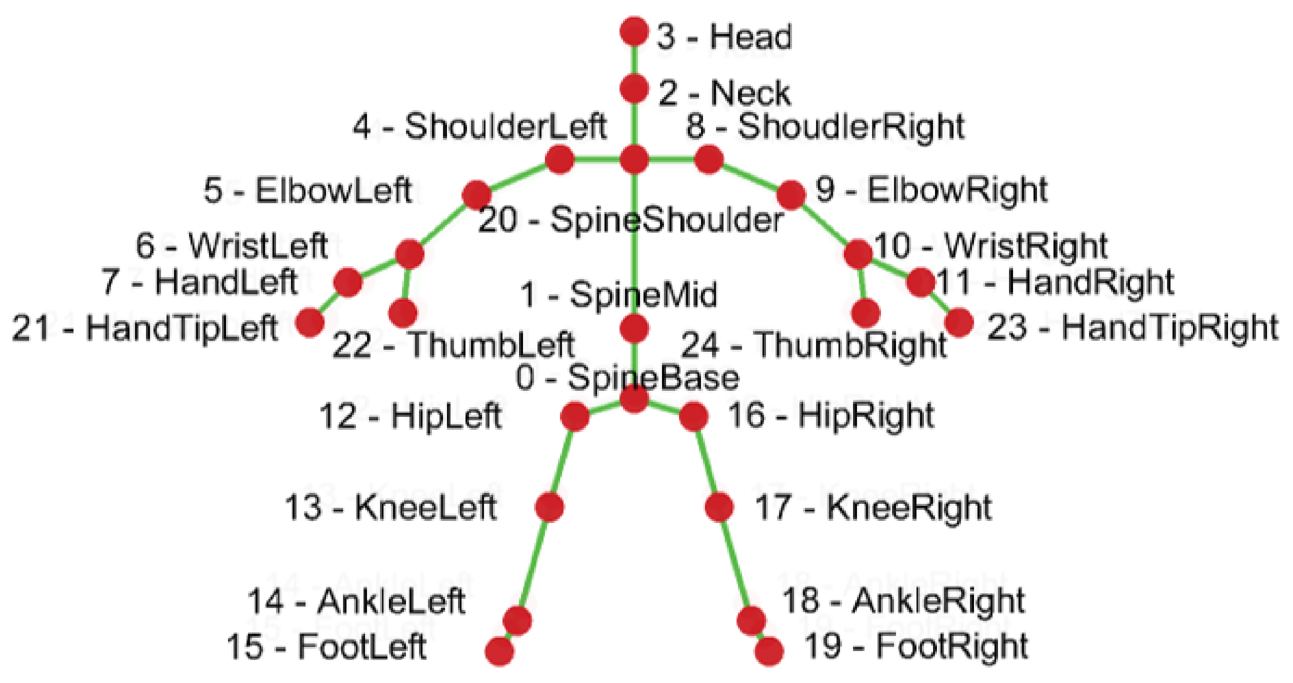

Figure 3: 3D-skeleton representation with 25 joints

We choose the activities for an industrial cobot application with activities that can be easily adapted to other applications, such as Human-Drone collaboration, researchers, etc. Our new dataset is composed of 10 activities (see Table 1), each activity made by 3 different actions.

Table 1: Creation of activities

Activity	Action 1	Action 2	Action 3
Starting (see figure 4)	Take off jacket	Sit down	Rub two hands
Leaving	Cross hands	Stand up	Put on jacket
Finalising	Writing	Point to something	Clapping
Searching for information	Put on glasses	Pick up	Type on keyboard
Celebrating	Stand up	Cheer up	Jump up
Dialing with somebody	Reach into pocket	Play with phone/ tablet	Phone call
Capitulating	Pick up	Reading	Tear up paper
Letting something down	Drop	Kicking something	Pick up
Undoing	Shake head	Tear up paper	Throw
Doing a break	Headache	Wipe face	Drink water

So, for each activity, there are 25 joints (x,y,z) and 240 frames. The temporal stage for each activity (so 3 consecutive actions) is thus represented as:

$$x_p = \left[(x_1)^1, (y_1)^1, (z_1)^1, (x_2)^1, \ldots, (x_{25})^{240}, (y_{25})^{240}, (z_{25})^{240}\right] \tag{2}$$

3.2 Data Pre-Processing

In order to reduce the training time, we decide to dwindle the number of input data of the neural network. One of the main features of Principal Component Analysis (PCA) [13] is the reduction of data. On each sequence, we extract the main components (1, 2, 3, etc) of the data with PCA. It is this feature that is used as input to the neural network. The temporal stage for each action for 1 component is represented as:

$$x_{pca} = \left[pc_{x1}, pc_{y1}, pc_{z1}, pc_{x2} \ldots pc_{z25}\right] \tag{3}$$

This drastically reduces the number of data (225 for each activities instead of 18.000).

For 2 components, we have:

$$\begin{aligned} x_{pca} = &[[pc_{1-x1}, pc_{1-y1}, pc_{1-z1}, \ldots pc_{1-z25}] \\ &\times [pc_{2-x1}, pc_{2-y1}, pc_{2-z1}, \ldots pc_{2-z25}]] \end{aligned} \tag{4}$$

The number of data for one activity is 450 if we choose 2 components.

3.3 Activity Recognition Using LSTM

The algorithm used in our work is a LSTM, because it has already shown its efficiency [17]. The input of the network is the joints skeleton or the PCA applied on the joints of each sequence. We followed this article [18] for creating our neural network. It is a composed of:

- a single LSTM hidden layer with 100 nodes,
- a layer for reducing the overfitting, which is a dropout layer with 50%,
- a dense layer with 100 features as output is used with ReLU as activation function for the interpretation of the features from the hidden layer,
- a dense layer with 10 output (for the number of different activities) with softmax as activation function.

The classification is realized with the categorical cross entropy loss function which is used for the multi-class classification. For the optimization of the network, the Adam version of the stochastic gradient descent is used [19].

The evaluation of the model is 5 times repeated because it is stochastic so it is possible to have different models as a result for the same input data. The final score is the summarize of the 5 different configurations. For the LSTM, the number of epochs is 100 and the batch-size is 100 before tuning.

The whole process of recognizing the activities represents thus a double-layer algorithm (PCA + LSTM as shown inFigure 5) which is different than the work done in [20] where the author concatenates 2 LSTM.

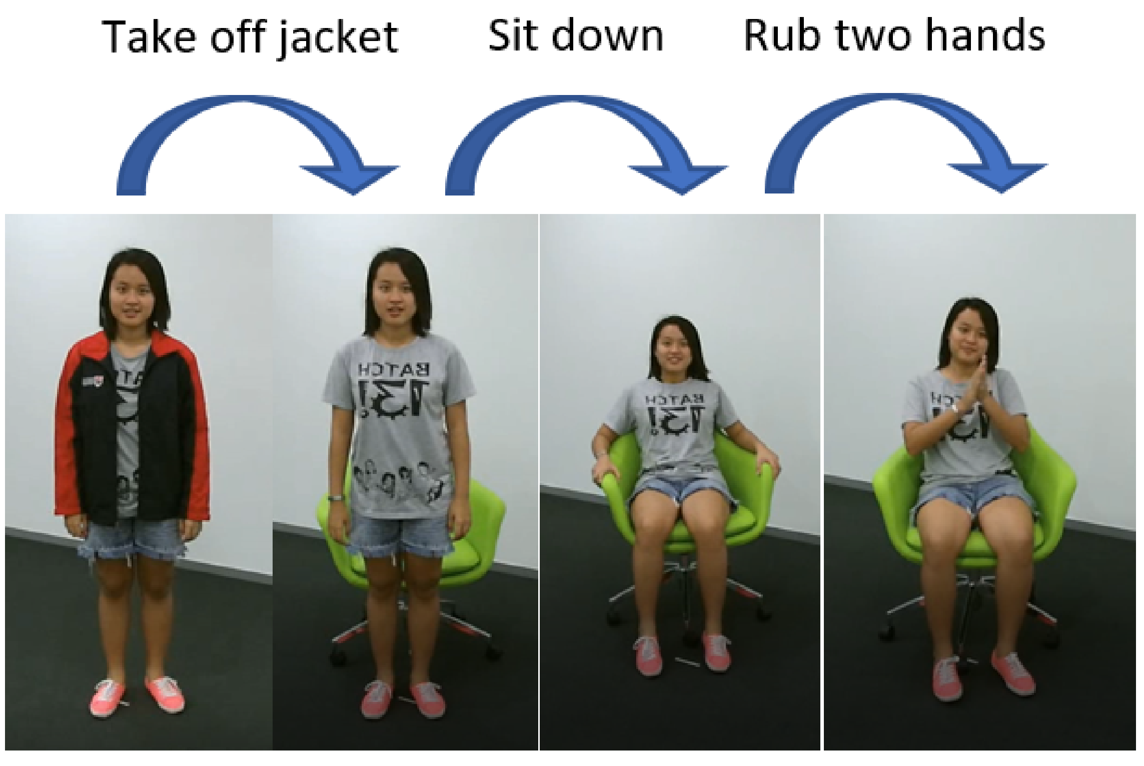

Figure 4: **Starting activity [2]**

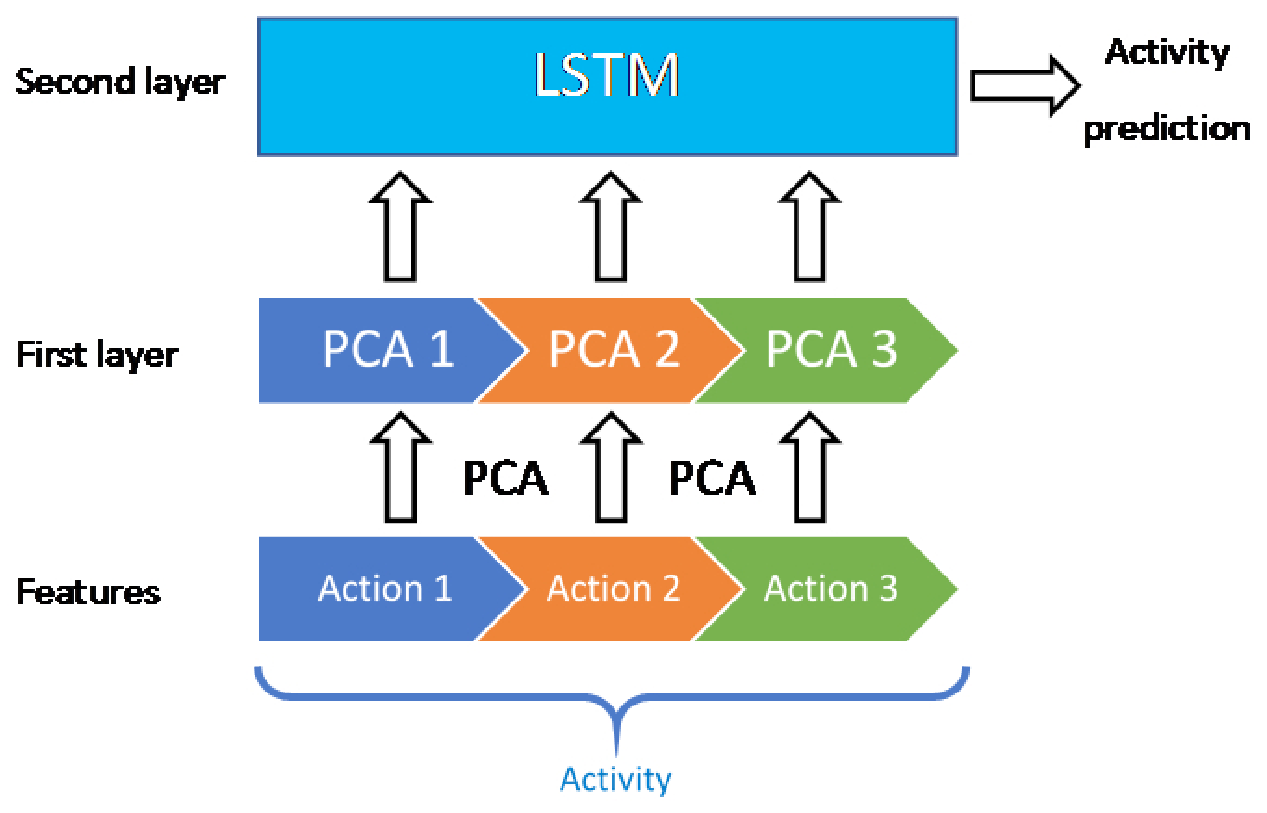

Figure 5: **Double-layer algorithm - PCA and LSTM**

4 EVALUATION

In this section we discuss the experiments made with a computer with an Intel Core i7 CPU of 1.8 GHz and 16 Go of RAM. We first tested our algorithm with the LSTM. Then, we applied the PCA with the number of components varying between 1 and 10 as feeding the LSTM.

4.1 Experimental Dataset and Protocol

For the experimental results, we used our proposed approach for activity recognition on the new dataset described in section 3. This database is an activity database composed of different collection of 3 actions.

The actions come from the NTU RGB+D dataset [2]. It is composed of 60 actions recorded by 3 cameras Kinect V2 providing RGB videos, depth map sequences, 25 joints of the 3D skeletal data and infrared videos. 40 performers are used for building the dataset and one person in present in each sequence. It represents 56,880 video samples. We only focus on the individual actions and remove the mutual actions. For training and testing, we follow the cross-subject protocol [2]. The training persons are : 1, 2, 4, 5, 8, 9, 13, 14, 15, 16, 17, 18, 19, 25, 27, 28, 31, 34, 35, 38. The remaining persons are used for the test. Furthermore, the empty files and the files with only one skeleton recorded are also removed. It considerably reduces the errors and increases the accuracy.

4.2 Results

For the first test, we use the LSTM algorithm on the 10 created activities without applying PCA. For this learning, 120 hours were needed and the accuracy was 95.533% (+/-0.573). The confusion Matrix is shown in figure 6. The use of another GPU could be considered to obtain faster results.

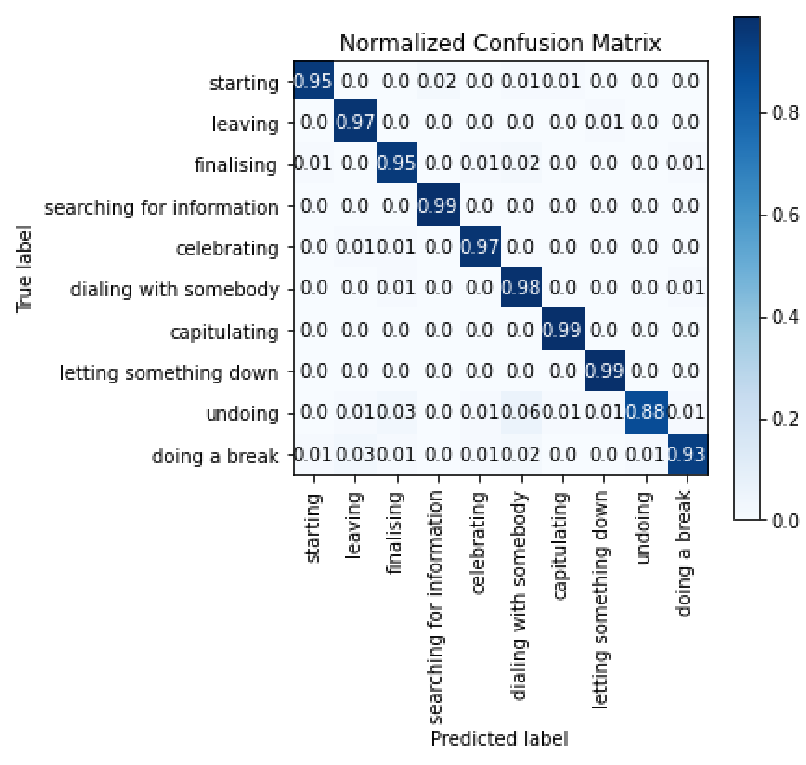

Figure 6: **Confusion Matrix for the activity recognition with simple-layer LSTM**

As a second stage, we evaluate the efficiency of the PCA applied before the LSTM algorithm by evolving the number of components of the PCA from 1 to 10. The Table 2 and figure 7 show the several accuracies according to the number of components. The results show for a PCA with a number of components over 1, the accuracy is higher than a classic LSTM. The 3 best rates are for n_components = 9, 8, 5 but for further work, we keep 5 because the training time and the number of data are the most reduced. The time used for learning with a 5 components is 8 hours, which is definitely smaller than the LSTM without PCA and which provides a better accuracy. Figure 8 shows the confusion Matrix for 5 components.

Table 2: LSTM and PCA-LSTM algorithm with n-components evolution

Method	Accuracy [%]
LSTM only	95.533 (+/-0.573)
PCA (n=9) + LSTM	96.826 (+/-0.383)

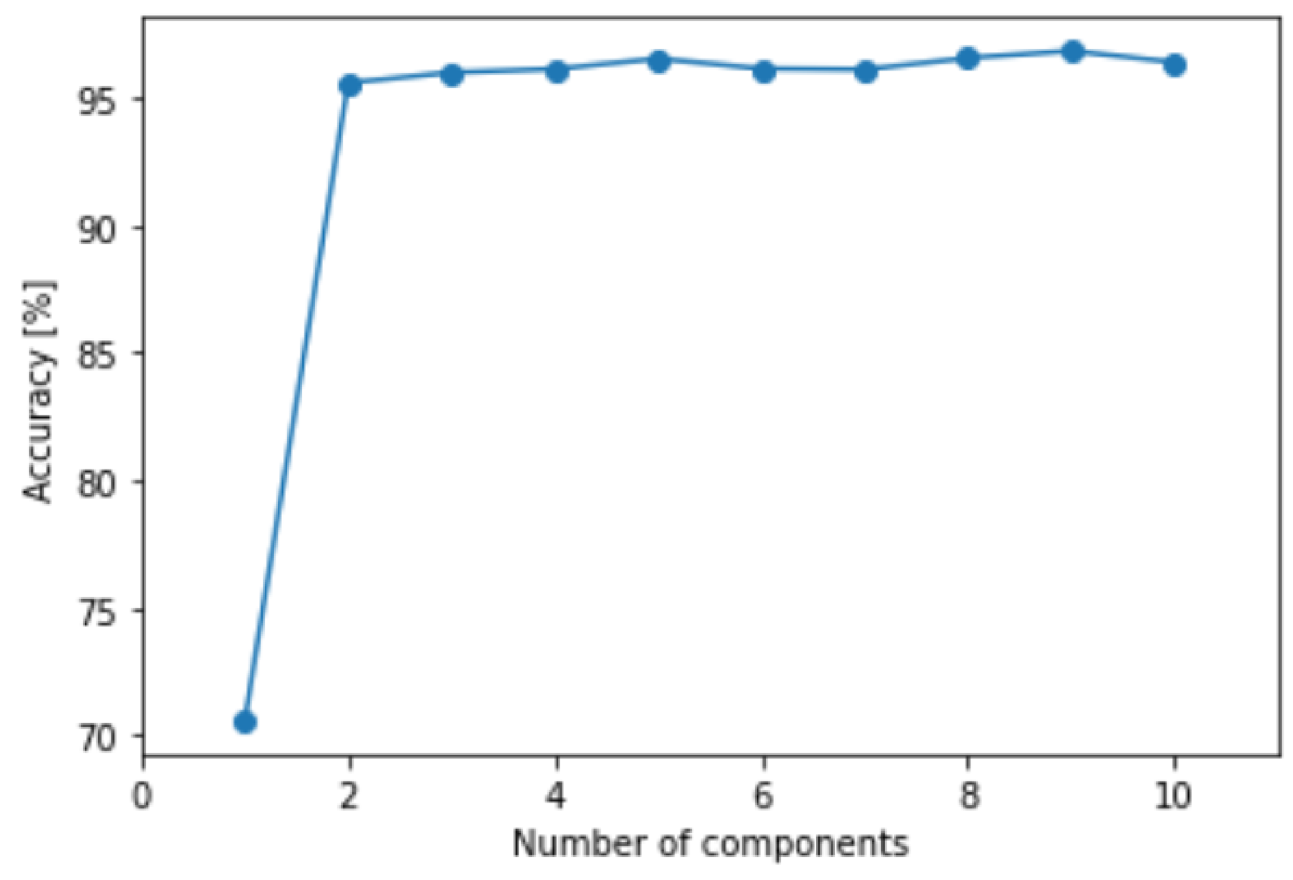

Figure 7: **Accuracy according to the variations of the number of components of the PCA-LSTM algorithm**

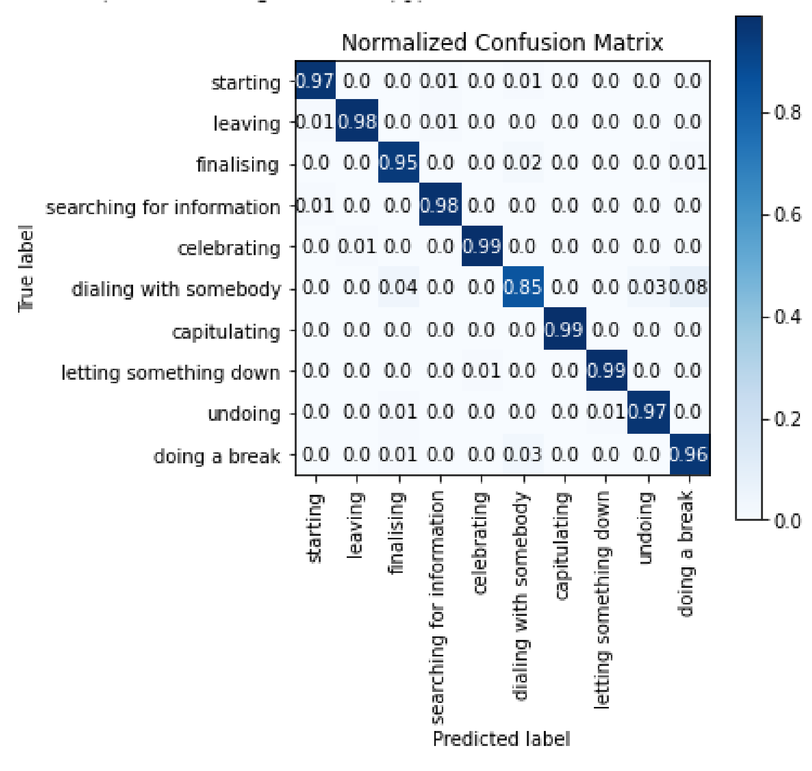

Figure 8: **Confusion Matrix for the activity recognition with our double-layer algorithm PCA + LSTM**

5 CONCLUSION

In this paper, we created a new activity dataset with the combination of three multiple actions from the NTU RGB+D dataset in the industrial environment. The goal of this dataset is the collaboration between a robot or a drone and a human-being. We tested the efficiency of this dataset with an LSTM and we tried the efficiency of the PCA with 1 to 10 components and realized it is efficient for PCA > 1 with a double-layer algorithm (PCA + LSTM). We concluded adding PCA for action pre-processing helped to reduce time and improved the results which is a real benefit when the input data size is large.

As a further work, we should extend the database with other activities. We can also use the NTU RGB+D 120 [21]. We also would like to use the sliding windows for evaluating the algorithm [22]. We can also focus on a few joints of the skeleton instead of taking the whole skeleton constituted of noisy joints such as fingers [20]. Finally, we would like to try our algorithm on a new activity dataset named Industrial Human Action Recognition Dataset (InHARD) [23].

ACKNOWLEDGEMENTS

This work was supported by Interreg and took part in the GRONE project at the technical department of the Henallux college located in Virton, Belgium.

REFERENCES

[1] Mark Murnane, Max Breitmeyer, Francis Ferraro, Cynthia Matuszek, and Don Engel. 2019. Learning from human-robot interactions in modeled scenes. In ACM SIGGRAPH 2019 Posters (SIGGRAPH '19). Association for Computing Machinery, New York, NY, USA, Article 1, 1–2. DOI:**https://doi.org/10.1145/3306214.3338546.Sam (https://doi.org/10.1145/3306214.3338546.Sam)** Anzaroot and Andrew McCallum. 2013. UMass Citation Field Extraction Dataset. Retrieved May 27, 2019 from **http://www.iesl.cs.umass.edu/data/data-umasscitationfield (http://www.iesl.cs.umass.edu/data/data-umasscitationfield)** Navigate to ⌄

[2] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, Gang Wang, "NTU RGB+D: A Large Scale Dataset for 3D Human Activity Analysis", IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016. Chelsea Finn. 2018. Learning to Learn with Gradients. PhD Thesis, EECS Department, University of Berkeley. Navigate to ⌄

[3] Song, Z.; Yin, Z.; Yuan, Z.; Zhang, C.; Chi, W.; Ling, Y.; Zhang, S. Attention-Oriented Action Recognition for Real-Time Human-Robot Interaction. arXiv 2020, **arXiv:2007.01065 (arXiv:2007.01065)**. Navigate to ⌄

[4] Akkaladevi, Sharath & Heindl, Christoph & Angerer, Alfred & Minichberger, Juergen. (2015). Action Recognition for Human Robot Interaction in Industrial Applications. 10.13140/RG.2.1.1069.2009. Navigate to ⌄

[5] Herath, S., Harandi, M., & Porikli, F. (2017). Going deeper into action recognition: a survey. Image and Vision Computing, 60, 4-21. **https://doi.org/10.1016/j.imavis.2017.01.010 (https://doi.org/10.1016/j.imavis.2017.01.010)**. Navigate to ⌄

[6] Varol, Gül & Laptev, Ivan & Schmid, Cordelia. (2018). Long-Term Temporal Convolutions for Action Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence. 40. 1510 - 1517. 10.1109/TPAMI.2017.2712608. Navigate to ⌄

[7] Johanna Carvajal, Chris McCool, Brian C. Lovell, and Conrad Sanderson. Joint recognition and segmentation of actions via probabilistic integration of spatio-temporal fisher vectors. CoRR, abs/1602.01601, 2016. Navigate to ⌄

[8] Yong Du, Wei Wang, and Liang Wang. Hierarchical recurrent neural network for skeleton based action recognition. In IEEE Int. Conf. on Computer Vision and Pattern Recognition, pages 1110–1118, 2015. Navigate to ⌄

[9] Hochreiter, Sepp & Schmidhuber, Jürgen. (1997). Long Short-term Memory. Neural computation. 9. 1735-80. 10.1162/neco.1997.9.8.1735. Navigate to ⌄

[10] Olah, C. (2015). Understanding LSTM Networks. Navigate to ⌄

[11] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, Yoshua Bengio Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling, CoRR, 2014. Navigate to ⌄

[12] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial Temporal Graph Convolutional Networks for Skeleton-Based Action Recognition. In AAAI, 2018. Navigate to ⌄

[13] Jamal, Ade & Handayani, Annisa & Septiandri, Ali & Ripmiatin, Endang & Effendi, Yunus. (2018). Dimensionality Reduction using PCA and K-Means Clustering for Breast Cancer Prediction. Lontar Komputer : Jurnal Ilmiah Teknologi Informasi. 192. 10.24843/LKJITI.2018.v09.i03.p08. Navigate to ⌄

[14] S. B. Dabhade et al., "Double Layer PCA based Hyper Spectral Face Recognition using KNN Classifier," 2017 International Conference on Current Trends in Computer, Electrical, Electronics and Communication (CTCEEC), Mysore, 2017, pp. 289-293, doi: 10.1109/CTCEEC.2017.8455113. Navigate to ⌄

[15] C. Wu, J. Zhang, S. Savarese and A. Saxena, "Watch-n-patch: Unsupervised understanding of actions and relations," 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, 2015, pp. 4362-4370, doi: 10.1109/CVPR.2015.7299065. Navigate to ⌄

[16] C. Wu, J. Zhang, O. Sener, B. Selman, S. Savarese and A. Saxena, "Watch-n-Patch: Unsupervised Learning of Actions and Relations," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 40, no. 2, pp. 467-481, 1 Feb. 2018, doi: 10.1109/TPAMI.2017.2679054. Navigate to ⌄

[17] S. Zhang, X. Liu and J. Xiao, "On Geometric Features for Skeleton-Based Action Recognition Using Multilayer LSTM Networks," 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), Santa Rosa, CA, 2017, pp. 148-157, doi: 10.1109/WACV.2017.24. Navigate to ⌄

[18] Guillaume Chevalier, LSTMs for Human Activity Recognition, 2016, **https://github.com/guillaume-chevalier/LSTM-Human-Activity-Recognition (https://github.com/guillaume-chevalier/LSTM-Human-Activity-Recognition)** Navigate to ⌄

[19] Diederik P. Kingma, Jimmy Ba: Adam: A Method for Stochastic Optimization, 3rd International Conference on Learning Representations, 2015. Navigate to ⌄

[20] Maxime Devanne, Panagiotis Papadakis, Sao Nguyen. Recognition of Activities of Daily Living via Hierarchical Long-Short Term Memory Networks. 2019. hal-02194928. Navigate to ⌄

[21] Jun Liu, Amir Shahroudy, Mauricio Perez, Gang Wang, Ling-Yu Duan, Alex C. Kot, "NTU RGB+D 120: A Large-Scale Benchmark for 3D Human Activity Understanding", IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 2019. Navigate to ⌄

[22] J. Yang, W. Liu, J. Yuan and T. Mei, "Hierarchical Soft Quantization for Skeleton-Based Human Action Recognition," in IEEE Transactions on Multimedia, doi: 10.1109/TMM.2020.2990082. Navigate to ⌄

[23] ALLEL, Mejdi, HAVARD, Vincent, BAUDRY, David, & SAVATIER, Xavier. (2020). InHARD - Industrial Human Action Recognition Dataset in the Context of Industrial Collaborative Robotics. IEEE International Conference on Human-Machine Systems (ICHMS). Navigate to ⌄

VSIP 2021, November 19–21, 2021, Wuhan, China

ACM ISBN 978-1-4503-8588-6/21/11...$15.00.
DOI: **https://doi.org/10.1145/3503961.3503966 (https://doi.org/10.1145/3503961.3503966)**