Human Action Recognition using Convolutional Neural Network: Case of Service Robot Interaction

Souhila Kahlouche[1] [a] and Mahmoud Belhocine[2] [b]
[1]*Ecole Nationale Supérieure d'Informatique (ESI), Oued Smar, Algiers, Algeria*
[2]*Centre de Développement des Technologies Avancées (CDTA), Baba Hassen, Algiers, Algeria*

Keywords: Human Robot Interaction (HRI), Human Activities Recognition (HAR), Deep Learning, Robot Operating System (Ros).

Abstract: This paper proposes a Human Robot Interaction (HRI) framework for a service robot capable of understanding common interactive human activities. The human activity recognition (HAR) algorithm is based on end to end deep Convolutional Neutral Network architecture. It uses as an input a view invariant 3D data of the skeleton joints, which is recorded from a single Microsoft Kinect camera to create a specific dataset of six interactive activities. In addition, an analysis of the most informative joint is made in order to optimize the recognition process. The system framework is built on Robot Operating System (ROS), and the real-life activity interaction between our service robot and the user is conducted for demonstrating the effectiveness of the developed HRI system. The trained model is evaluated on an experimental dataset created for this work and also the publicly available datasets Cornell Activity Dataset (CAD-60), and KARD HAR datasets. The performance of the proposed algorithm is proved when compared to other approaches and the results confirm its efficiency.

1 INTRODUCTION

Social robots must be able to interact efficiently with humans, to understand their needs, to interpret their orders and to predict their intentions. This can be achieved by translating the sensed human behavioural signals and context descriptors into an encoded behaviour. However, it stills a challenge because of the complex nature of the human actions. Even if many researches related to human-robot interaction (HRI) were announced, there were not so many reports of its successful application to robotic service task.

Limin in (Limin, M., & Peiyi, Z., 2017) used a Kinect camera to capture human actions in real time. They used this information to send commands to the robot through the Bluetooth communication and make some movements as turning and forward. Borja in (Borja et al., 2017) presented an algorithm, which used depth images from Kinect to control the speed and angular position of a mobile robot. The algorithm used the previous frame for the segmentation of the current one, thus, a user should extend the hand in front of the camera and leaves it for a while until the program recognizes the hand. They also designed a PID to control the wheel speed of the robot.

In the context of robot assisted living, Zhao in (Zhao et al., 2014) proposed a gesture recognition algorithm for taking order service of an elderly care robot. It was designed mainly for helping non-expert users like elderly to call a service robot. Faria in (Faria et al., 2015), designed a skeleton-based features model, and used it on mobile robot in a home environment, to recognize daily and risky activities in real-time and to react for assisting the person.

For Human Action Recognition task, the early well-known approaches are hand-crafted based features. In these approaches researchers extract features such as global or local image features like texture, edge or other attributes from the frames and use them within different machine learning algorithms (Yang, X., & Tian, Y., 2014), (Suriani 2018) and (Kahlouche, S., & Belhocine, 2019).

[a] https://orcid.org/0000-0001-8353-4566
[b] https://orcid.org/0000-0003-3495-7444

Kahlouche, S. and Belhocine, M.
Human Action Recognition using Convolutional Neural Network: Case of Service Robot Interaction.
DOI: 10.5220/0011122300003271
In *Proceedings of the 19th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2022)*, pages 105-112
ISBN: 978-989-758-585-2; ISSN: 2184-2809

However, hand-crafting features require significant domain knowledge and careful parameter tuning, which makes it not robust to various situations.

Recently, with the emergence and successful deployment of deep learning techniques for image classification, researchers have migrated from traditional handcrafting to deep learning techniques for HAR. Several works employed Convolutional Neural network CNN, due to its spectacular progress in extracting spatial correlation characteristics in image classification tasks (Simonyan, K., & Zisserman, 2014) and (Hascoet, 2019).

We presented in previous works (Bellarbi, A et al., 2016), and (Kahlouche, S et al. 2016), ROS based framework for B21r service robot, including a social navigation approach where the robot was able to navigate in indoor environment while detecting and avoiding humans using several social rules, which are computed according to its skeleton pose orientations. Additionally, a Guide User Interface (GUI) has been designed to control the robot through its screen touch which displays some functionality such as Call the Robot, Follow Me and Guide Me.

In this works, we aim to make the Human-Robot interaction more natural and more intuitive, by integrating HAR module to our framework. This is an alternative solution to the GUI use. We proposed a deep learning architecture for activity recognition to help the robot to recognize which service is requested by the user, and react consequently.

For this purpose, we investigate the usefulness of using only 3D skeleton coordinates of human body in end-to-end deep CNN to learn the spatial correlation characteristics of human activities. Hence, the mains considered contributions are:

- Data transformations in the pre-processing step applied on the 3D skeleton data, in order to reduce observation variations caused by viewpoint changes.
- An analysis of the Most Informative Joints has been done to evaluate the informativeness of each joint in the dataset, which leads to feature dimensionality reduction when training the CNN on relevant data.

2 HRI SYSTEM OVERVIEW

Figure 1 shows our proposed architecture block diagram, it uses mainly offline and online process.

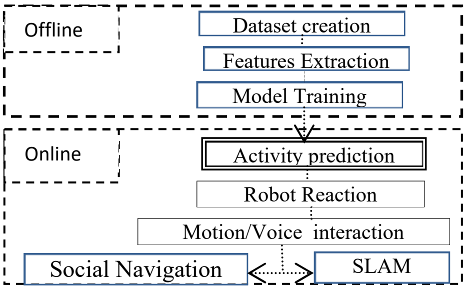


Figure 1: The framework of a human-robot interaction.

2.1 Dataset Creation

Our dataset has been created using OpenNI tracker framework, which allows the skeleton tracking at30 fps, providing 3D Euclidean coordinates and three Euler angles of rotation in the 3D space for each joint with respect to the sensor. The dataset contains six interactive activities performed by four different individuals. Among the activities, four are static, where the user does not move from the camera view field (Hello, Stop, Call, Pointing), and two dynamic activities where the user can leave the camera view field (Going, Coming). A seventh class named "No Activity" has been added, it is realized with sequences where the user is immobile or when he does not express any interaction gestures.

Figure 2 shows some examples of our dataset:

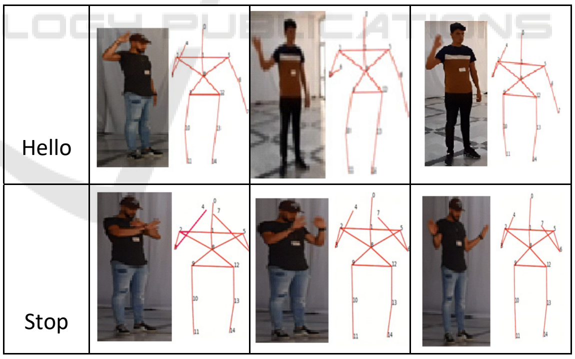


Figure 2: Examples of our dataset.

2.2 Features Extraction

In most activities, many joints contribute very little changes and are not significant for action recognition. Hence, an analysis is performed to concentrate only on significant joints which highly contribute to human activity. According to (Ofli et al., 2014) works called Sequence of Most Informative Joints (SMIJ) reported that Shannon entropy represents the higher entropy related to a maximum contribution of relative joints. Additionally, for Gaussian distribution with variance σ^2 the entropy can be calculated by the formula of Equation (1).

$$H(I) = \frac{1}{2}(\log 2\pi\sigma^2 + 1) \tag{1}$$

With: $$\sigma = \sqrt{\frac{(I-\bar{I})^2}{N}} \tag{2}$$

I : joint position,

$\bar{I}$: the average joint position

N: the number of significant implications of joint I in a given temporal sequence.

Figure 3 represents the histogram of the most informative joints for six (6) types of activities in our dataset. Therefore, we have selected five significant joints in the human skeleton and applied suitable weight on them. On the other hand, we have ignored the remainder by setting the joints to zero, since they introduce more noise than useful information to distinguish activities.

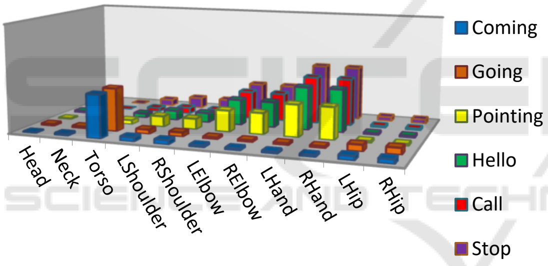


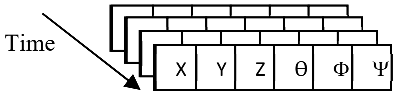
Figure 3: Most Informative Joints of our dataset.

2.2.1 Construction of Feature Vectors

Using the above information, we can compute a set of features as follows:

- 3D coordinates of the 5 selected joints: (x,y,z) which are all relative to the torso;
- 3D rotation angles in the space: Ψ ,Θ and Φ a sequence of three rotations according to the three axes X, Y and Z respectively.

Therefore, the feature vector has the following form:



2.2.2 Feature Pre-Processing

A pre-processing step is applied on the 3D skeleton data in order, not only to attenuate noise introduced by the sensor, but also to normalize the data to *accommodate* for different users' heights, limb lengths, orientations and positions. It consists of the following steps:

- *Translation*: to guarantee the same origin of the coordinates system for all acquired frames, the reference is set to the torso of the human skeleton;
- *Normalization*: to reduce the influence of different users' heights and limb lengths, first, the height of the subject is determined; then all skeleton 3D coordinates are normalized according to the value of eq.3;

$$D_{i-normal} = \frac{D_i - \min(D_i)}{\max(D_i) - \min(D_i)} \tag{3}$$

- *Symmetrization*: To disambiguate between mirrored versions of the same activity (e.g. gestures performed by right and left-handed people), it is required for activities such as Hello, Call and pointing. It is just necessary to consider a new sample based on a mirrored version of the original 3D skeleton data.

2.3 Model Training

We have used CNN algorithm Network for training with17 layers. Figure 4 presents the overall structure of this pipeline.

Input Layer: It is a vector representing sub video sequences of the 3D skeleton data:

$N_{attributes} \times N_{joint} \times N_{frames}$ .Where $N_{attributes}$is a vector composed of $x \times y \times z \times \Theta \times \Phi \times \Psi$. While N_{joint} is the number of joints associated with each configuration in the video sequence of the dataset and it is equal to 5. N_{frames} which is the total number of frames in batch sequences and it is equal to 30; Zero padding is added to this layer.

Convolutional Layer1: The input layer is scanned using 32filters of size 3x3. Batch normalization is used to accelerate the training of the networks; it consists of performing the normalization for each training mini-batch. The used activation is Rectified Linear Unit (ReLU).

Max Pooling1: It is useful to capture the most important features and reduce the computation in advanced layer, we have used (2,1) pooling and zero padding.

Convolutional Layer2: 64 filters of size 5x5, ReLU function and batch normalisation are used;

Convolutional Layer3: 128 filters of size (3,3), with batch normalization and ReLU;

Convolutional Layer4: 256 filters of size (3,3), with batch normalization and ReLU;

Max Pooling Layer2: We have used 2x2 kernels and 0 padding;

Convolutional Layer5: 256 filters of size (3, 3), batch normalization, and Elastic-Net (1e-4) regularization which help for lessening over fitting.

Max Pooling3: Kernel size (2,2), and stride of (2,2), 0 padding;

Convolutional Layer6: 256 filters of size (3,3), 10% Dropout, and ReLU is applied as activation function;

Max Pooling4: Kernel size (2, 2) and 10% Dropout are applied;

Convolutional Layer7: 256 filters of size (3,3), zero padding, 10% Dropout, *Elastic-Net* (1e-4) regularization and ReLU is applied as activation function;

Flattened Layer: Concatenation of 256 feature vectors into vector of one dimension containing 1024 features.

Fully Connected Layer1: it produces 256 neurons from the last layer, batch normalization, 30% Dropout, Regularization *Elastic-Net* (1e-4) and ReLU are used;

Fully Connected Layer2: 128 neurons with batch normalization, ReLU and 30% Dropout are used;

Fully Connected Layer3: 64 neurons, with batch normalization, ReLU, 30% Dropoutand *Elastic-Net* (1e-3) regularization are used;

Output Layer: At the end, it has7 classes with Softmax

We have used, for training, the Keras deep learning framework with a Tensor Flow backend on a laptop with an i5-2320 (3.00GHz) CPU. The network has been trained using Adam optimizer, and Categorical Cross Entropy as loss function. After several attempts of parameter tuning, the best results are obtained at epoch 2000 using a learning rate of 10-5 with an initial training rate set to 0.001, a batch size of 30, 0 padding and stride (1, 1) for all convolution layers.

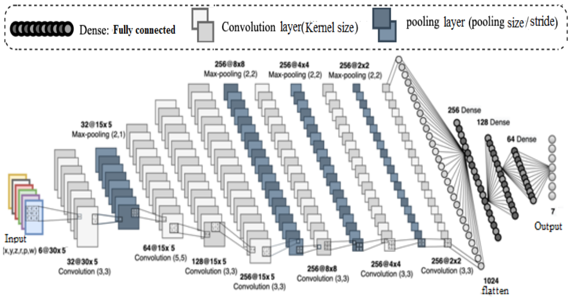


Figure 4: The overall structure of the pipeline.

2.3.1 Performance on Collected Dataset

Our dataset is challenging because of the following reasons:

(i) High interclass similarity: some actions are very similar to each other, for example, Coming/Going, and Hello/Call.

(ii) High intra-class variability: the same action is performed in different ways by the same subject. For example, using left, right, or both hands differently.

(iii) The activity sequences are registered from different views.

Our dataset has been divided into two parts; 75% for training and 25% for testing.

The performance of HAR has been evaluated based on the accuracy percentage of activities that are correctly recognized. The result shown in the confusion matrix achieves accuracy of 97.42**%** (Figure5).

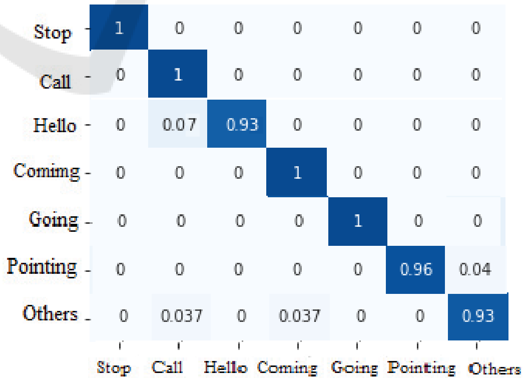


Figure 5: Confusion matrix.

2.4 The Online Processing

- Human Activity Prediction package: The developed HAR module has been implemented under Robot Operating System (ROS), to recognize the performed activity by the service robot.

- Autonomous Social Navigation package: This is built from ROS navigation stack, given the current locations of obstacles; it uses the global planner to find a path to a desired destination. It then uses a local planner to compute linear and angular velocities that need to be executed by the robot to approximately follow the global path while avoiding obstacles and humans differently respecting some social rules (Bellarbi , A, et al., 2016).

- Simultaneous Localization and Mapping (SLAM): we used Hector Slam package, which provides robot position and an environment map, for self-localization.

Once the activity is recognized by the prior step, the appropriate reaction is taken from the look-up table to be executed by the mobile robot in real interaction scenarios (Table1).

The robot can perform motion or voice reaction; for the voice reaction, we have used a speech synthesis module to play sound from a given input text.

Table 1: Six common types of interaction.

Activity	Robot reaction
Hello	*Voice reaction* : Hello welcome to CDTA
Call	*Voice reaction* : Please wait, I am coming *Motion reaction:* Approaching user and start interaction.
Stop	*Voice reaction:* Ok, I will stop here. *Motion:* Stop moving.
Pointing	*Voice reaction:* Ok, I will go there. *Motion reaction*: Go to the pointed position.
Coming	*Voice reaction*: How can I help you? *Motion reaction*: Step back and prepare to begin interaction.
Going	*Voice reaction*: Good-bye, thank you for your visit. *Motion*: turn back and stop interaction

3 RESULTS AND DISCUSSIONS

3.1 Performance on Public Datasets

We present a comparative performance evaluation on two public datasets: the CAD-60 and KARD HAR datasets.

The *CAD-60* dataset (Sung, J, et al. 2012) is performed by 12 different activities, typical of indoor environments that are performed by four different people: two males and two females.

The KARD dataset (Gaglio, S. et al. 2014) consists of 18 activities. This dataset has been captured in a controlled environment, that is, an office with a static background, and a Kinect device placed at a distance of 2-3m from the subject. The activities have been performed by 10 young people (nine males and one female), aged from 20 to 30 years, and from 150 to 185cm tall. Each person repeated each activity 3 times in order to create 540 sequences. The dataset is composed of RGB and depth frames. Additionally, 15 joints of the skeleton in world and screen coordinates are provided.

Table2 lists the results of the proposed method, which are applied on the above mentioned datasets. Despite its simplicity, it is able to achieve good results when applied to publicly available datasets

Table 2: Comparison of our method on different dataset.

Dataset	Accuracy
KARD	95.0%
CAD60	83.19%
Our dataset	97.42**%**

Table 3 shows a comparison of the proposed method with other state of the art approaches on CAD-60. It can be seen from Table3 that despite very good accuracy obtained with different methods in the last decade, accuracy of the proposed activity recognition approach outperforms existing approaches. The proposed system achieved an accuracy of 83.19 %.

Table 3: Comparison of accuracy on CAD60 dataset.

Algorithm proposed by	Accuracy
Zhu et al. (2014)	62.5%
Yang et al.(2014)	71.9 %
Rahmani et al. (2014)	73.5%
Zhang et al.(2012)	81.8 %
Wang et al.(2013)	74.7%
Gaglio et al. (2014)	77.3%
Koppula et al.(2013)	80.8%
Nunes et al. (2017)	81.8 %
Proposed approach	**83.19%**

3.2 Performance on Mobile Robot

In order to properly test the system in real scenarios, we have used the B21r mobile robot, which is able to map the indoor environment and to self-localizing and autonomously navigating while avoiding obstacles and humans. The developed HRI system has been implemented under Robot Operating System (ROS). For real time prediction, we used the last recorded sequence during one second, according to the sensor's refresh rate (30 fps) after being pre-processed. After that, a final decision is made for activity recognition and a robot reaction is performed.

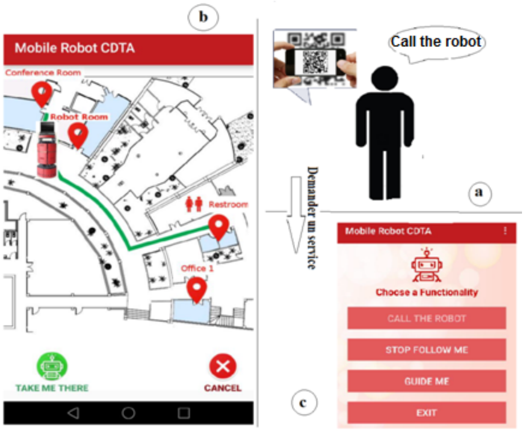


Figure 6: Older scenario with GUI.

Figure 6 shows the older scenario of HRI:

a) To call the robot, the user has to scan using his Smartphone, the QR code associated to its real word position.

b) The real world map, where some predefined positions in the environment are predefined such as: Conference room, Library, Entrance, Robot room. Hence, the robot must generate an appropriate collision free trajectory and navigate until reaching its goal.

c) The Guide User Interface GUI to control the robot through its screen touch which displays some functionality such us: Call the robot, Follow Me, Guide Me.

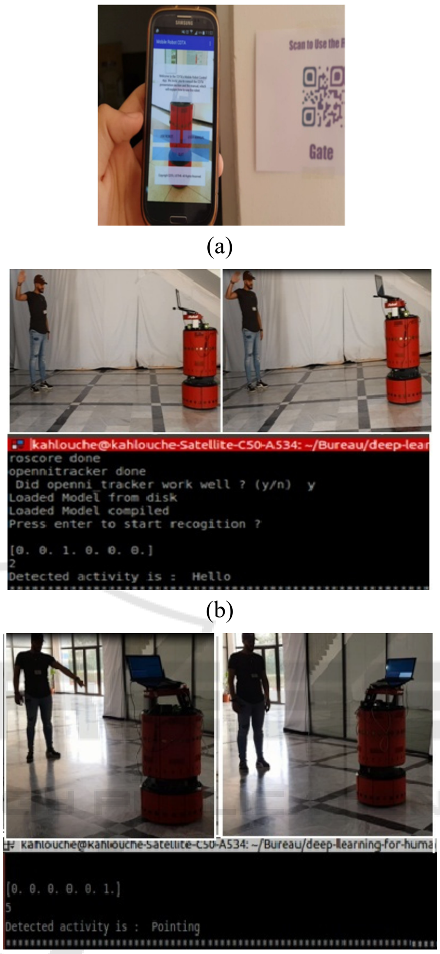
Figure 7 shows the new scenario where a person attempts to interact with a service robot:

a) First, the person Calls the robot from its distant initial position (robot room) by scanning the QR code associated to its position using its smart phone.

b) The robot navigate until the user position, detect the person and start interaction with him, the user salutes the robot, 'Hello' action is recognized and the robot react with voice mode and say: 'Hello, welcome to our Center'.

c) The user is pointing to a specific position; the activity is recognized as "pointing", and the voice reaction of the robot is: "Ok, I will go there", and the robot move to the pointed position.

The proposed framework was capable of recognizing different interactive activities that happen sequentially in case of a person transits from one activity to another.



(a)

```
kahlouche@kahlouche-Satellite-C50-A534: ~/Bureau/deep-lear
roscore done
opennitracker done
 Did openni_tracker work well ? (y/n)  y
Loaded Model from disk
Loaded Model compiled
Press enter to start recogition ?

[0. 0. 1. 0. 0. 0.]
2
Detected activity is :  Hello
```

(b)

```
kahlouche@kahlouche-Satellite-C50-A534: ~/Bureau/deep-learning-for-huma

[0. 0. 0. 0. 0. 1.]
5
Detected activity is :  Pointing
```

c)

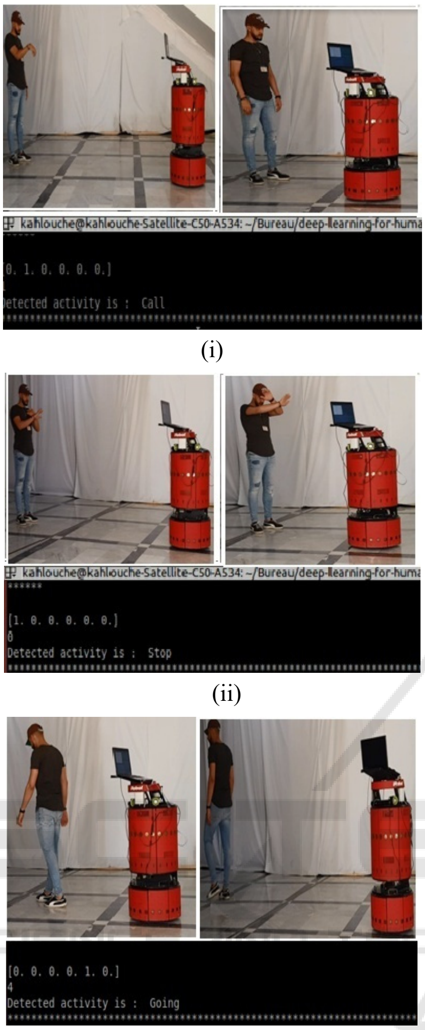
Figure 7: Natural interaction scenario: Distant Call – Hello-Pointing.

Figure 8 shows a second natural interaction scenario where:

i) The person is calling the robot in its proximity, and the activity is recognized as "Call" and then the robot interacts with voice mode, and say: "*Please wait, I am coming*", while approaching the user.

ii) The user decides to stop the robot; the activity is recognized as "Stop", and the robot stop moving, and say: "ok, I will stop here".

iii) The user decides to go away, the activity "Going" is recognized and the robot react with voice: "Good- bye, thank you for your visit", and turn back and stop interaction.



```
kahlouche@kahlouche-Satellite-C50-A534: ~/Bureau/deep-learning-for-huma
******

[0. 1. 0. 0. 0. 0.]
1
Detected activity is :  Call
*****************************************************************
```

(i)

```
kahlouche@kahlouche-Satellite-C50-A534: ~/Bureau/deep-learning-for-huma
******

[1. 0. 0. 0. 0. 0.]
0
Detected activity is :  Stop
*****************************************************************
```

(ii)

```
[0. 0. 0. 0. 1. 0.]
4
Detected activity is :  Going
*****************************************************************
```

(iii)

Figure 8: Scenario b) Proximity Call – Stop – Going.

4 CONCLUSION

We presented a Human Robot Interaction system for a service robot, able to recognize the performed activity and to successfully react according to the situations. In the pre-processing step, we have proposed a view invariant transformation applied to the data, captured by a Microsoft Kinect camera to guarantee view invariant features. To achieve features dimensionality reduction, an analysis of the most informative joints has been performed while concentrating on significant joints which highly contribute to the human activity. Therefore, five significant joints in the human skeleton have been selected and used as input layer of a deep CNN. This model has been tested in real time successfully, hence, it presents a promising approach to social robotics field where natural and intuitive human robot interaction is needed. However, some further developments are needed before the system can be used in a real life. Therefore in the future, we want to consider the use of other sensor modalities such as depth maps and RGB sequences in order to add additional contextual information and see what the best architecture to fuse all these modalities is. This should improve the activity recognition accuracy and consequently will improve the interactivity of our service robot.

REFERENCES

Bellarbi, A., Kahlouche, S., Achour, N., & Ouadah, N. (2016, November). A social planning and navigation for tour-guide robot in human environment. In *2016 8th International Conference on Modelling, Identification and Control (ICMIC)* (pp. 622-627). IEEE. https://doi.org/10.1109/ICMIC.2016.7804186.

Borja, J. A. T., Alzate, E. B., & Lizarazo, D. L. M. (2017, October). Motion control of a mobile robot using kinect sensor. In *2017 IEEE 3rd Colombian Conference on Automatic Control (CCAC)* (pp. 1-6). IEEE.

Faria, D. R., Vieira, M., Premebida, C., & Nunes, U. (2015, August). Probabilistic human daily activity recognition towards robot-assisted living. In *2015 24th IEEE international symposium on robot and human interactive communication (RO-MAN)* (pp. 582-587).

Gaglio, S., Re, G. L., & Morana, M. (2014). Human activity recognition process using 3-D posture data. *IEEE Transactions on Human-Machine Systems*, *45*(5), 586-597.

Hascoet, T., Zhuang, W., Febvre, Q., Ariki, Y., & Takiguchi, T. (2019). Reducing the Memory Cost of Training Convolutional Neural Networks by CPU Offloading. *Journal of Software Engineering and Applications*, *12*(8), 307-320. doi: 10.4236/jsea.2019.1 28019.

Kahlouche, S., & Belhocine, M. (2019, November). Human Activity Recognition Based on Ensemble Classifier Model. In *International Conference on Electrical Engineering and Control Applications* (pp. 1121-1132). Springer, Singapore. https://doi.org/10.1007/978-981-15-6403-1_78

Kahlouche, S., Ouadah, N., Belhocine, M., & Boukandoura, M. (2016, November). Human pose recognition and tracking using RGB-D camera. In *2016 8th International Conference on Modelling, Identification and Control (ICMIC)* (pp. 520-525). IEEE. DOI: 10.1109/ICMIC.2016.7804168.

Koppula, H. S., Gupta, R., & Saxena, A. (2013). Learning human activities and object affordances from rgb-d videos. *The International journal of robotics research*, *32*(8), 951-970.

Limin, M., & Peiyi, Z. (2017, March). The medical service robot interaction based on kinect. In *2017 IEEE International Conference on Intelligent Techniques in Control, Optimization and Signal Processing (INCOS)* (pp. 1-7). IEEE.

Nunes, U. M., Faria, D. R., & Peixoto, P. (2017). A human activity recognition framework using max-min features and key poses with differential evolution random forests classifier. *Pattern Recognition Letters*, *99*, 21-31.

Ofli, F., Chaudhry, R., Kurillo, G., Vidal, R., & Bajcsy, R. (2014). Sequence of the most informative joints (smij): A new representation for human skeletal action recognition. *Journal of Visual Communication and Image Representation*, *25*(1), 24-38.

Rahmani, H., Mahmood, A., Du Huynh, Q., & Mian, A. (2014, September). HOPC: Histogram of oriented principal components of 3D pointclouds for action recognition. In *European conference on computer vision* (pp. 742-757). Springer, Cham.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Sung, J., Ponce, C., Selman, B., & Saxena, A. (2012, May). Unstructured human activity detection from rgbd images. In *2012 IEEE international conference on robotics and automation* (pp. 842-849). IEEE.

Suriani, S., Noor, S., Ahmad, F., Tomari, R., Nurshazwani, W., Zakaria, W. W., & Mohd, M. H. (2018). Human activity recognition based on optimal skeleton joints using convolutional neural network. *Journal of Engineering Science and Technology*, *7*, 48-57.

Wang, J., Liu, Z., Wu, Y., & Yuan, J. (2013). Learning actionlet ensemble for 3D human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, *36*(5), 914-927.

Yang, X., & Tian, Y. (2014). Effective 3d action recognition using eigenjoints. *Journal of Visual Communication and Image Representation*, *25*(1), 2-11.doi.org/10.1016/j.jvcir.2013.03.001

Zhang, C., & Tian, Y. (2012). RGB-D camera-based daily living activity recognition. *Journal of computer vision and image processing*, *2*(4), 12.

Zhao, X., Naguib, A. M., & Lee, S. (2014, August). Kinect based calling gesture recognition for taking order service of elderly care robot. In *The 23rd IEEE international symposium on robot and human interactive communication* (pp. 525-530). IEEE.

Zhu, Y., Chen, W., & Guo, G. (2014). Evaluating spatiotemporal interest point features for depth-based action recognition. *Image and Vision computing*, *32*(8), 453-464.