Exploring Artificial Intelligence methods for recognizing human activities in real time by exploiting inertial sensors

Dimitrios Boucharas*, Christos Androutsos*, Nikolaos S. Tachos*, *Member, IEEE*, Evanthia E. Tripoliti†, *Senior Member, IEEE*, Dimitrios Manousos‡, Vasileios Skaramagkas‡, Emmanouil Ktistakis‡, Manolis Tsiknakis§, *Member, IEEE*, Dimitrios I. Fotiadis*†, *Fellow, IEEE*

*Dept. of Biomedical Research, FORTH-IMBB, Ioannina, Greece

†Unit of Medical Technology and Intelligent Information Systems,

Dept. of Materials Science and Engineering, University of Ioannina, Ioannina, Greece

‡ Institute of Computer Science, FORTH, Heraklion, Greece

§ Dept. of Electric and Computing Engineering, Hellenic Mediterranean University, Heraklion, Greece

Abstract—The aim of this work is to present two different algorithmic pipelines for human activity recognition (HAR) in real time, exploiting inertial measurement unit (IMU) sensors. Various learning classifiers have been developed and tested across different datasets. The experimental results provide a comparative performance analysis based on accuracy and latency during fine-tuning, training and prediction. The overall accuracy of the proposed pipeline reaches 66% in the publicly available dataset and 90% in the in-house one.

Index Terms—Machine Learning, Deep Learning, HAR, Sensor

I. INTRODUCTION

Over the last decade, an increasing interest in human activity recognition (HAR) has emerged. Several methods have been leveraged to tackle this challenging classification problem. It is a crucial yet open issue that needs to be addressed to assist the eldercare and healthcare. Due to technological advances, HAR may be performed by utilizing the sensory ecosystem of wearable devices and smartphones, captured images or a video data stream [1] [2].

HAR is a problem that tries to predict different activities based on sensor data derived from human motion. It is a time series classification problem which tries to map activities with associated human movement or action. Movements and actions are performed during walking, jogging, talking and sitting but more specific movements can exist while cooking or handcrafting [3]. Therefore, sensors can translate these movements and actions into numerical time series data. The mapping can be utilized between features extracted from the sensor data and activities or raw sensor data and activities. The first is utilized on machine learning (ML) methods while the latter on deep learning (DL) ones.

This work provides a comparison analysis which is based on time series acquisition data from Inertial Measurement Unit (IMU) sensors. From the literature, Zhan *et al.* [2]

proposed an activity recognition system that collects time series data and video from an accelerometer and a first person camera, respectively. Features were extracted from both sensor signal and frame data in order to train two ML classifiers. A local and a structured model were implemented to provide predictions on human activities. The first depends on the raw sensor data, while the other considers temporal dependencies to capture contextual relations. Possas *et al.* [4] implemented a Recurrent Neural Network (RNN) architecture which is the Long Short-Term Memory (LSTM) architecture, to classify human activities relied on motion sensor data and the Long-term Recurrent Convolutional Network (LRCN) on vision data. In an attempt to balance these two different techniques, reinforcement learning policies have been exploited.

In the work presented by Song *et al.* [5], the Fisher Kernel framework was applied to fuse features from sensors and video recordings. Later, Song *et al.* [6] developed a multi-stream Convolutional Neural Network (CNN) to captivate characteristics from videos. In parallel, multi-stream LSTM was integrated to learn features from accelerometer and gyroscope. The last two networks were fused along with various pooling techniques. Arabacı *et al.* [7] focused on an adaptive framework that weighed visual, audio and sensor features related to their ability to distinguish human activities. They compared Kernel SVM, MKBoost and SimpleMKL algorithms which resulted in Polynomial SVM yielding the best performance.

In this paper, a comparison between conventional ML and a DL technique is presented. Not only the performance in widely acclaimed accuracy metrics is taken into account, but also in the parameter tuning, the training and the prediction (inference) times. Therefore, they should both be weighed according to the needs of the implementation. Finally, a publicly available dataset and a newly in-house collected one

have been utilized to evaluate the classifiers.

II. MATERIALS AND METHODS

A. Dataset

Three datasets have been exploited in order to develop the proposed scheme: DataEgo, DataEgo* and an in-house dataset.

DataEgo is an egocentric dataset composed of visual and inertial data. This dataset consists of 20 activities which can be further split into 5 groups: mobility, office work, kitchen related, exercising and general. The data was captured with the Vuzix M300 Smart Glasses. Each session lasts around 5 minutes and mimics a real-world scenario that corresponds to a sequence of 4 to 6 different activities. By analyzing the sensor information, some variability can be noted for a given activity. The data includes activity annotation per sample. Each sample consists of acceleration and gyroscope information. Each sensor provides values from the corresponding three-axis coordinate system. The sampling rate of the accelerometer and gyroscope is 15 Hertz. This dataset is highly imbalanced. It should be noted that the minority class contains 796 observations while the majority has more than 31000.

DataEgo* is a subset of DataEgo which contains the classes related to high physical motion activities such as walking, cycling, doing push-ups, running and doing sit-ups and some activities that belong to the lower physical motion like eating and reading. DataEgo* was generated in order to overcome the high imbalance of DataEgo.

To tackle the limited amount of public datasets and further assess the generalization of the developed models, a newly collected dataset was needed. After the approval from the ethical committee, volunteers, at the IMBB-FORTH, Ioannina, Greece, were asked to take part in a data collection study and perform a set of activities wearing a portable sensor ecosystem. The ecosystem integrated a pair of glasses which had a mounted IMU sensor. The sensor utilized was the MetaMotionR, from which the accelerometer and gyroscope data were collected. This dataset consists of 10 participants. Each participant performed a sequence of activities, yet some of which performed a second one. The activity sequence mimics a real-world scenario of activities of daily living (ADL). The high motion activities performed are sitting, standing, walking, upstairs and downstairs. The participants were asked to maintain their personal style and speed when doing an activity and act as natural as possible. The chosen time spent per activity was predetermined to provide a balanced dataset. The sample rate of the accelerometer and gyroscope data is 50 Hertz. Accelerometer and gyroscope range lies in ± 16 g and ± 2000 dps, respectively. The total number of collected samples is 649301 which corresponds to 3.6 hours of annotated dataset. Each sample contains six values and the annotated class. The values represent the accelerometer and gyroscope information. Annotation was based on a timestamp application that captured activity changes and on the graph representation of the inertial data.

B. Hardware

To develop the in-house dataset, a 3D printed pair of glasses was utilized. A 9-axis inertial measurement unit sensor (MetaMotionR) was mounted on top of them. The processing unit (LattePanda A864s) and the sensor communicate via Bluetooth Low Energy. Additionally, a portable powerbank of 30.000 mAh was attached as the power supply. As a result, the glasses became a portable wearable system which is shown in Fig. 1.

Fig. 1: Hardware components.

C. Methodology

For the classification of ADL, conventional ML and DL techniques have been developed. The classifiers that have been utilized are the Random Forests (RFs), the Support Vector Machines, the k-Nearest Neighbors (k-NN) and an Artificial RNN architecture, the Long Short-Term Memory. SVMs exploit the ability of a dataset to be separable by a hyperplane. K-NN predicts the class based on k neighbors according to a distance function. RFs, combine several decision trees in order to provide the most common inference. Lastly, LSTMs exploit both feedforward and feedback connections in its neural network architecture along with its ability to learn and forget.

The developed ML scheme employs a sliding window method to augment the data. The window size is 200 data points with an overlapping ratio 80%, which corresponds to 4 seconds. For each window, a feature vector is extracted manually. It includes features calculated not only per axis but also a combination of them. Mean, kurtosis, energy and signal magnitude area are some of the features calculated. In total, 208 features are extracted. To support the performance of the algorithms, a min-max normalization is applied, which scales the values of the features in $[0, 1]$. However, some features do not hold the same level of importance as others. For that reason, analysis of variance (ANOVA) and chi-squared feature ranking algorithms have been employed. Whether the model parameters are derived from an accuracy plot, an error plot or a grid search, tuning is necessary.

The LSTM architecture is able to learn and recall over long multiple parallel sequences of input data provided by the axes of accelerometer and gyroscope sensor. The model learns to extract features from sequences of observations and then maps the internal features to different types of activity. The benefit of using LSTM for time series classification is that it has the ability to learn from raw data directly and retain the information for a long period of time by design.

Fig. 2: Methodological Pipeline.

To improve the performance of the developed DL scheme, raw data were scaled in $[-1, 1]$ integrating min-max normalization. LSTM expects sequences of fixed-length training data. Thus, the sensor signal is separated into fixed-size time windows with an overlap of 90%. The window size is 200 data points corresponding to 4 seconds. The network takes batches of time series as input with a batch size 500. The architecture implemented consists of 2 LSTM stacked layers with 128 hidden units each and 2 fully connected layers. Two different optimizers have been tested, Stochastic Gradient Descent and Adam optimizer. The number of epochs used for the training phase is 300. The proposed workflow is depicted in Fig. 2.

III. RESULTS

The models are trained and tested in the datasets described in Section II.A. At the same time the 10-fold stratified cross validation (SCV) is utilized as an evaluation method for all datasets based on the mean accuracy of the folded classifiers. Additionally, leave – one sequence – out validation (LOSO) is implemented for the in-house dataset.

For the DataEgo dataset, the accuracy of the SVM classifier reached 33.32%, while SVM with polynomial kernel of third degree (Poly-SVM), RF and k-NN achieved 40.34%, 44.26% and 43.25%, respectively. SVM with Radial Basis kernel Function (RBF-SVM) achieved 66.75% reporting the highest performance. By utilizing ANOVA and Chi-Squared feature selection techniques, the above-mentioned classifiers provided a lower accuracy score by a margin of 4 – 7% in all cases. Despite the lower scores, the model was simpler and converged faster. It should also be noted that parameter tuning time for RBF-SVM was larger comparing to the rest. Specifically, the grid search took 18 hours to complete, while the tuning of the rest was performed in a couple of minutes. LSTM achieved 45.75%, and can be further increased if more data is available.

For the DataEgo* dataset, the accuracy of conventional SVM reached 68.91%, while Poly-SVM, RF and k-NN achieved 72.10%, 77.15% and 74.15%, respectively. RBF-SVM achieved 84.09%, maintaining its position as top ML

TABLE I: Comparison of Machine and Deep Learning classifiers.

Classifier	Accuracy	
	DataEgo	DataEgo*
SVM	33.32%	68.91%
Poly-SVM	40.34%	72.10%
RF	44.26%	77.15%
k-NN	43.25%	74.15%
RBF-SVM	66.75%	84.09%
LSTM	45.75%	58.22%

choice. By utilizing ANOVA and chi-squared feature selection methods, the classifiers provided a lower yet closer than before accuracy score by a margin of 1–4% in all cases with about a third of the features. Margins are noted to be shrunk. LSTM yielded a score 58.22%. The results of the classifiers are summarized in Table I.

Due to the high performance of RBF-SVM, it is considered the best choice from the ML algorithms. The accuracy metric is derived from the mean accuracy of classifiers. RBF-SVM managed to reach 90.82% on SCV and 90.72% on LOSO. LSTM achieved 84.83% on SCV, 85.31% on LOSO, which further enriched the alignment that more data were needed for such a model. The classifier results from the median accuracy of the folds are depicted in Fig. 3.

Fig. 3: Confusion Matrix from LSTM LOSO.

The most notable error is observed on misclassification of “sitting” and “standing”. During these activities, no unique movements noticed. Therefore, a motion sensor cannot perceive these movements and the corresponding numerical data of these activities end up similar. Another misinterpretation occurs in “walking” and “upstairs”, which might be a case of the type of stairs during the data collection protocol. The stairs had a walking part on each floor.

It was also observed that the precision, the recall and the f1 scores are all within 0–3% difference from the accuracy score in all experiments. Parameter tuning is only used for the radial basis function, while the parameters for LSTM have been selected based on experiments. Thus, it is a

demanding task that requires high computational power and time. Training time on the other hand, is a far less time-consuming assignment. As for the prediction time of SVM, it is nearly instantaneous in contrast with LSTM which requires about 4.5 seconds. Overall, it was observed that both approaches achieved good accuracy levels. Despite the superiority presented by RBF-SVM in the training and prediction time, parameter tuning prohibits online training. LSTM seems to provide a higher level of automation but a slower inference. The comparison details are shown in Table II.

TABLE II: Comparison of the proposed real time human activity recognition methods.

Metrics		RBF-SVM	LSTM
Accuracy	SCV	90.82%	84.83%
	LOSO	90.72%	85.31%
Parameter Tuning Time		18 hrs	-
Training Time		0.85 mins	27.7 mins
Prediction Time		78 ms	4601 ms

IV. DISCUSSION

The proposed activity recognition models utilize sensor data streams, derived from accelerometer and gyroscope, to differentiate high motion magnitude ADL. The proposed algorithmic pipeline is validated by utilizing the wearable system of the See Far solution [8]. The See Far solution includes smart glasses, based on augmented and ML technologies to provide the end users, that suffer from a specific vision impairment, with a plethora of supportive functionalities. The proposed models are integrated in the See Far personalized profile.

Similar approaches have been explored in the literature by Zhan *et al.*, Possas *et al.*, Song *et al.* and Arabacı *et al.*. For the publicly available datasets, Multimodal and DataEgo dataset, the accuracy reached up to 58% and 64%, respectively. The Multimodal dataset is no longer publicly available however it resembles the DataEgo dataset. Differences are mainly on the visual information.

TABLE III: Comparison with literature.

Authors	Dataset	# Activities	Accuracy	
Zhan <i>et al.</i> [2]	In-house	12	66%	
Possas <i>et al.</i> [4]	DataEgo	20	58%	
	Multimodal	20	61%	
Song <i>et al.</i> [5] [6]	Multimodal	20	49%	
Arabacı <i>et al.</i> [7]	Multimodal	20	64%	
Proposed Approach	RBF-SVM	DataEgo	20	66%
		DataEgo*	7	84%
		In-house	5	90%
	LSTM	DataEgo	20	45%
		DataEgo*	7	58%
		In-house	5	85%

The proposed ML approach reaches 66% accuracy in the DataEgo dataset. Thus, it exhibits better performance than Possas *et al.*, Song *et al.* and Arabacı *et al.* work. ML and DL schemes achieve better performance in the in-house

dataset due to the high physical motion activities being executed. IMU sensors can capture specific movements or actions during these activities. In contrast, Zhan *et al.* in-house and DataEgo datasets include low motion activities, such as reading or working on pc, that do not indicate any special movement for a head motion sensor. As a result, accuracy was increased for DataEgo*, which includes mostly high motion activities. To the same extent, accuracy was significantly increased for the in-house dataset achieving up to 90%. This comparison is depicted in Table III.

V. CONCLUSION

In this work, two algorithmic pipelines have been designed and developed for HAR in real time. The latter is part of the personalized profile of the See Far solution. More specifically, it is implemented into the See Far augmented reality smart glasses which assist vision impaired population with services and recommendations in real time. The developed models can reach up to 90% accuracy for the in-house dataset while yielding great results in the publicly available datasets. As future work, a wider variety of activities will be explored without influencing the overall performance.

VI. ACKNOWLEDGMENT

Research supported by the See Far project [8], which has received funding from the European Union's Horizon 2020 research and innovation program under the grant agreement No 826429. This article reflects only the authors' view. The European Commission is not responsible for any use that may be made for the information it contains.

REFERENCES

- [1] Vavoulas, George and Chatzaki, Charikleia and Malliotakis, Thodoris and Padiaditis, Matthew and Tsiknakis, Manolis, "The mobiact dataset: Recognition of activities of daily living using smartphones", *International Conference on Information and Communication Technologies for Ageing Well and e-Health*, vol. 2, pp. 143–151, 2016.
- [2] Zhan, Kai and Faux, Steven and Ramos, Fabio, "Multi-scale conditional random fields for first-person activity recognition", *2014 IEEE international conference on pervasive computing and communications (PerCom)*, pp. 51–59, 2014.
- [3] Lara, Oscar and Labrador, Miguel, "A survey on human activity recognition using wearable sensors", *Multimedia Tools and Applications*, vol. 15, no. 3, pp. 1192–1209, 2012.
- [4] Possas, Rafael and Caceres, Sheila Pinto and Ramos, Fabio, "Ego-centric activity recognition on a budget", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5967–5976, 2018.
- [5] Song, Sibio and Chandrasekhar, Vijay and Mandal, Bappaditya and Li, Liyuan and Lim, Joo-Hwee and Sateesh Babu, Giduthuri and Phyo San, Phyo and Cheung, Ngai-Man, "Multimodal multi-stream deep learning for egocentric activity recognition", *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 24–31, 2016.
- [6] Song, Sibio and Cheung, Ngai-Man and Chandrasekhar, Vijay and Mandal, Bappaditya and Liri, Jie, "Egocentric activity recognition with multimodal fisher vector", *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2717–2721, 2016.
- [7] Arabacı, Mehmet Ali and Özkan, Fatih and Surer, Elif and Jančovič, Peter and Temizel, Alptekin, "Multi-modal egocentric activity recognition using multi-kernel learning", *Multimedia Tools and Applications*, vol. 80, no. 11, pp. 16299–16328, 2021.
- [8] See Far project: Available: <https://www.see-far.eu/>