

2018

## A lightweight classification algorithm for human activity recognition in outdoor spaces

Graham McCalmont

*Ulster University, School of Computing, Coleraine, United Kingdom*

Huiru Zheng

*Ulster University, Belfast, Coleraine, United Kingdom*

Haiying Wang

*Ulster University, School of Computing, Coleraine, United Kingdom*

*See next page for additional authors*

Follow this and additional works at: <https://arrow.tudublin.ie/teapotcon>



Part of the [Electrical and Electronics Commons](#), and the [Systems and Communications Commons](#)

---

### Recommended Citation

McCalmont, G., Zheng, H. & Wang, H. (2018). A lightweight classification algorithm for human activity recognition in outdoor spaces. *Proceedings of the 32nd International BCS Human Computer Interaction Conference, Belfast, UK, 4 - 6 July*. doi:10.14236/ewic/HCI2018.53

*This Conference Paper is brought to you for free and open access by the tPOT: People Oriented Technology at ARROW@TU Dublin. It has been accepted for inclusion in Conference Papers by an authorized administrator of ARROW@TU Dublin. For more information, please contact [arrow.admin@tudublin.ie](mailto:arrow.admin@tudublin.ie), [aisling.coyne@tudublin.ie](mailto:aisling.coyne@tudublin.ie), [vera.kilshaw@tudublin.ie](mailto:vera.kilshaw@tudublin.ie).*

---

**Authors**

*Graham McCalmont, Huiru Zheng, Haiying Wang, S. I. McClean, Matteo Zallio, and Damon Berry*

# A Lightweight Classification Algorithm for Human Activity Recognition in Outdoor Spaces

Graham McCalmont, Huiru Zheng\*, Haiying Wang, Sally McClean  
School of Computing, Ulster University,  
Jordanstown Campus, Shore Road,  
Newtownabbey, Co. Antrim, BT37 0QB, Northern Ireland,  
{mccalmont-g, h.zheng\*, hy.wang, si.mcclean}@ulster.ac.uk

Matteo Zallio, Damon Berry  
Dublin Institute of Technology,  
Kevin Street, Dublin,  
D08 X622, Ireland  
{matteo.zallio, damon.berry}@dit.ie

**The aim of this paper is to discuss the development of a lightweight classification algorithm for human activity recognition in a defined setting. Current techniques to analyse data such as machine learning are often very resource intensive meaning they can only be implemented on machines or devices that have large amounts of storage or processing power. The lightweight algorithm uses Euclidean distance to measure the difference between two points and predict the class of new records. The results of the algorithm are largely positive achieving accuracy of 100% when classifying records taken from the same sensor position and accuracy of 80% when records are taken from different sensor positions. The outcome of this work is to foster the development of lightweight algorithms for the future development of devices that will consume less energy and will require a lower computational capacity.**

*Lightweight algorithm, Activity recognition, Euclidean distance, Machine learning, Energy efficiency, Low-computational capacity*

## 1. INTRODUCTION

Development in the Internet of Things means the world is quickly filling up with digital sensors measuring everything from location, movement to humidity [1]. This is increasing the amount of data generated in the world. Estimates show that the amount of data generated is doubling every two years [1]. However, to unlock the full potential of all of this data it needs to be analysed. If the data goes unanalysed it is arguably just noise.

Current techniques for data analysis such as machine learning or artificial intelligence are excellent at making sense of all this data. However, this analysis comes with a high computational cost. This high amount of computation can require large amounts of storage or processing power. Machine learning also has the drawback of needing the model to be trained on a data set before accurate results can be given. Collecting enough data to train a model is often time consuming and expensive to complete [2].

It is important that we give users the ability to analyse all this data. With 68% of UK adults owning a smartphone it is likely a smartphone is the most common computational device owned [3]. But these high cost techniques like machine learning are too computationally intensive to be implemented well (or at all) on a smartphone or smartwatch [2]. Current methods to allow a user to analyse this data with a

smartphone often involve accessing some external resource rich service or platform over the internet. This in turn requires the user to have an internet connection. Connecting to the internet is still an issue for many users given that 4G mobile data coverage is limited to just 43% of the UK's landmass [4]. Wi-Fi is another option, but this is mostly limited to indoor spaces and large Wi-Fi networks cause some pretty serious security concerns, so it is not an ideal option.

It therefore makes sense that we try to perform some local analysis of the data near the point of data collection on the smart phone. This will enable users to interact fully with the data around them. A possible solution is the development of lightweight algorithms. Such algorithms would have a low computational cost and therefore would not need the computing power of an external resource and could be performed on a smartphone without the need to connect to the internet. This paper will present an example of a low computational algorithm for activity recognition in an outdoor gym.

The paper will be structured as follows; section two will explain the data set and the algorithm. Section three will present the results of the algorithm and discuss the findings comparing them to classification results from an ANN (artificial neural network). Finally, section four will conclude the paper.

## 2. METHODOLOGY

The data used in this study was collected as part of a wider study, in collaboration with the Dublin Institute of Technology, on the development of a Bluetooth enabled MEMS device for increasing user engagement in outdoor areas. The data set consists of nine subjects each performing two exercises with an air walker and pull-down machine, in an outdoor location.

While performing the exercises each of the participants was wearing three smartphones located at the upper arm, wrist and hip pocket. Each of the smartphones contains a gyroscope and accelerometer. The before mentioned MEMS device was also attached to the gym equipment and recorded each of the exercises. Resulting in sixty-nine records being collected. (a protocol issue with one of the files means only sixty-eight records are used in the tests).

The goal of the light weight algorithm will be to classify which of the two exercises is being performed, regardless of what position (hip pocket, wrist, upper arm or machine mounted) the data is recorded at. The classification accuracy will be compared to the results of a simple ANN trained and tested on the same data set.

### 2.1 Euclidean classification algorithm (ECA)

The lightweight algorithm uses Euclidean distance to measure how different records are from each other. Euclidean distance is defined as the shortest path between two points. This is not a new measurement by any standard. It has been used in the development of a fuzzy motion classifier [5]. Classification is performed by establishing example profiles for each of the exercises, more information on how the example profile are selected is given in section 2.2.

The distance from a new point to each of the example profiles gives a good indication as to which class the new record belongs to. Whichever path has the shortest distance means it is likely to be part of that class.

The algorithm reads in data from the sensors gyroscope (X,Y and Z) and accelerometer (X,Y and Z). From the accelerometer data total acceleration is calculated using the Equation (1). Where  $a_x$  is acceleration in the X direction,  $a_y$  is acceleration in the Y direction and  $a_z$  is acceleration in the Z direction. To reduce the noise from the data windowing is applied, taking the mean of every five data points. From this data we extract fourteen features listed in Table 1. At this stage all 14 features are used, though they may not all be required, feature ranking and selection will be the next step in the development of the algorithm (see further developments in conclusion). The example profiles also contains the same fourteen features and noise reduction is applied in the same way.

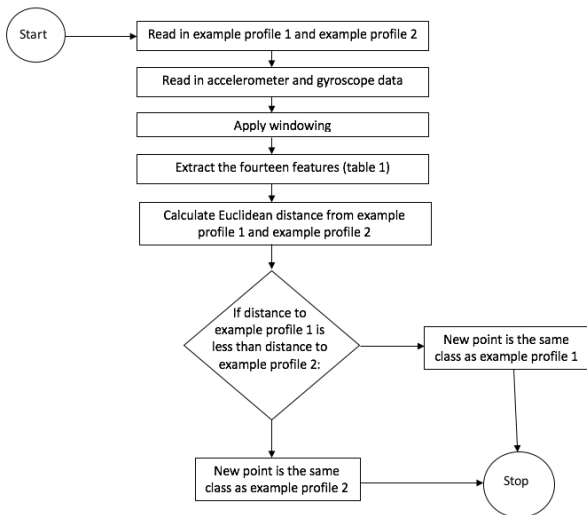
$$TA = \sqrt{a_x^2 + a_y^2 + a_z^2} \quad (1)$$

We then calculate the Euclidean distance from a new point to an example profile using equation (2) Where  $d$  is the distance,  $n$  is the number of features,  $p$  is the new point containing the fourteen features outlined in Table 1 and  $c$  is an example profile containing the same fourteen features.

$$d(p, c) = \sqrt{\sum_{i=1}^n (p_i - c_i)^2} \quad (2)$$

**Table 1:** name and description for the fourteen features in the feature vector

| Feature Number | Feature Name | Feature Description                                   |
|----------------|--------------|---|
| 1              | AvTA         | Mean of total acceleration                            |
| 2              | SdTA         | Standard deviation of total acceleration              |
| 3              | AvAx         | Mean of acceleration in x direction                   |
| 4              | AvAy         | Mean of acceleration in y direction                   |
| 5              | AvAz         | Mean of acceleration in z direction                   |
| 6              | AvGx         | Mean of angular velocity in x direction               |
| 7              | AvGy         | Mean of angular velocity in y direction               |
| 8              | AvGz         | Mean of angular velocity in z direction               |
| 9              | SdAx         | Standard deviation of acceleration in X direction     |
| 10             | SdAy         | Standard deviation of acceleration in y direction     |
| 11             | SdAz         | Standard deviation of acceleration in z direction     |
| 12             | SdGx         | Standard deviation of angular velocity in x direction |
| 13             | SdGy         | Standard deviation of angular velocity in y direction |
| 14             | SdGz         | Standard deviation of angular velocity in z direction |



**Figure 1:** Flow diagram from Euclidean classification algorithm

We then measure the distance from the new point to the other example profile. Whichever gives the shorter distance is the predicted class of that new data point. The algorithm has been summarised in the flow diagram shown in Figure 1

### 2.2 Selection of the example profiles

The selection of an example profile for each of the exercises is one of the most important parts of the algorithm. If the point is too central to its own class, it might not classify outliers correctly. It would be very simple to run the algorithm multiple times to find the two records from the given data set that gives the best results. However, this would be an unfair test of the algorithm as the results are not likely to be that good in a real-world situation. Instead what is a better and fairer test would be to find which of the four-sensor positions is best at classifying records from the other sensors. Two records for this sensor group can then be used as the example profile. This will be a fairer test of the algorithm and give a truer reflection of how it would cope in the real world.

To find out which was the best sensor position, four tests were conducted. First, we selected a random test subject. The data from this test subject was used as the example profile for each of the tests.

The four tests were structured as follows; test how well the upper arm data from the random test subject classified the upper arm data from the other eight test subjects this was repeated for hip pocket, wrist and the machine mounted data.

To compare these tests the accuracy, specificity and sensitivity for the four tests has been calculated for each of the classes using Equations (3)-(5). Where TP is true positive, TN is true negative, FP is false positive, and FN is False negative. The results for the four tests are shown in Table 2

$$Acc(\%) = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \quad (3)$$

$$Sp(\%) = \frac{TN}{TN + FP} \times 100 \quad (4)$$

$$Se(\%) = \frac{TP}{TP + FN} \times 100 \quad (5)$$

The results for these tests show that the hip pocket is the worst at predicting the class of records taken from the same sensor position. This is probably due to the fact that the hip pocket is less secure than the upper arm, wrist and machine mounted positions. It is very easy for the smartphone to be placed in the pocket upside down or back to front in the pocket. Meaning the accelerometer and gyro-scope would give very different results for each of the records and it would be hard to classify them.

**Table 2:** Accuracy, specificity and sensitivity for each of the sensor groups

|                 | Accuracy (%) | Pull-down       |                 | Air walker      |                 |
|-----------------|--------------|-----------------|-----------------|-----------------|-----------------|
|                 |              | Specificity (%) | Sensitivity (%) | Specificity (%) | Sensitivity (%) |
| Arm Data        | 93           | 85              | 100             | 100             | 85              |
| Hip pocket Data | 46           | 57              | 37.5            | 37.5            | 57              |
| Wrist data      | 100          | 100             | 100             | 100             | 100             |
| Machine Mounted | 86           | 75              | 100             | 100             | 75              |

As for which sensor position performed the best, it is clear that the phone located at the wrist manages to sense 100% of the records correctly. This is followed closely by upper arm and the machine mounted sensor. From this it is reasonable to suggest that wrist will be the best position where to locate a sensor to classify further records.

In a real-world situation an application would ask the user to perform the exercise, so it could take a recording to use as an example profile. This would allow the profile to be tailored to an individual user.

### 3. RESULTS AND DISCUSSION

This section will present the results for classification of exercise using the ECA. The results will be compared to the results of an artificial neural network trained and tested with the same data set. As discussed in the methodology section, wrist records from test subject one was selected for the example profiles for the two exercises.

The algorithm has an accuracy of 77% which is not as good as expected but there is a clear explainable reason for this. The wrist is actually not the best sensor to select for the example profiles. If the upper arm data, which came second in classification accuracy of its own sensor group, is used as example profiles, then accuracy would increase up to 80%, which represents an interesting improvement of the system.

An artificial neural network is trained and tested on the same data set. The data set was split into a training and testing set. Due to the small size of the data set and to achieve a fair distribution of each of the classes and sensor locations in the test set a ratio of 80:20 was used. The accuracy achieved by the network was 90%. The accuracy, specificity and sensitivity (calculated by Equations (3), (4) and (5)) for these three experiments can be seen in Table 3.

The results show the ANN performs better than the ECA when classifying records from different sensor positions. However, the ECA has no training overhead so it can be still considered as a simpler computational algorithm. More work will be

performed for improving the algorithm to investigate if the accuracy can be improved.

While the ECA didn't cope as well when classifying records from different sensor positions it is able to correctly classify 100% of records from data taken from the wrist so it would be ideal for a smart watch. The accuracy from the machine-mounted sensor and the upper arm are also above 85% (see Table 1) which could also be used in real world situations.

As demonstrated in the paper, success of the algorithm largely depends on how well the example profile describes the exercise. The method presented in this paper achieved good results. However new methods could be used to get better results for example where the average of several records could be calculated to develop a more centralised profile.

The goal of this paper was to develop a computationally efficient algorithm. The ECA can be considered computationally efficient for different reasons. When compared to machine learning techniques calculations in the ECA are relatively simple. Machine learning approaches also require training before they can be considered useful. This requires large amounts of data. On the other hand, in this paper has also been considered training and testing a ANN. For this part of the process 80% of the data had to be used to train the network. Whereas the ECA only requires one record for each of the classes in the problem, this is a significant reduction in the data used.

The algorithm performs well when classifying records from the same sensor position, but the computational cost of the ECA algorithm could be lower. Further research in this area could consider whether the noise reduction is required. Are all fourteen features required to get accurate results? Answering any of these questions could allow us to reduce the computational cost of the algorithm even further.

**Table 3:** Accuracy, Specificity and sensitivity for the multiple sensor location tests and the ANN

|   | Accuracy (%) | Pull-down       |                 | Air walker      |                 |
|---|--------------|-----------------|-----------------|-----------------|-----------------|
|   |              | Specificity (%) | Sensitivity (%) | Specificity (%) | Sensitivity (%) |
| Using wrist record as example profile     | 77           | 90              | 64              | 64              | 90              |
| Using upper arm record as example profile | 80           | 84              | 76              | 76              | 84              |
| ANN results                               | 90           | 100             | 83              | 83              | 100             |

#### 4. CONCLUSION

This paper presented a lightweight algorithm that was able to classify what exercise is being performed, if data from the same sensor is used. While its ability to classify records from different sensor positions is not as good as originally expected.

In the following stages of the project it is planned that the accuracy in classifying different gym equipment and the sensitiveness in detecting the human activity recognition associated can be improved.

The next step in this line of research should be to investigate if we can reduce the computational cost of the algorithm further while improving the accuracy of the algorithm and in a further step how energy efficiency of these devices can be further increased in accordance with a recent trend in technology development called "Planned Obsolescence" [5].

#### 5. Acknowledgements

The authors of this paper would like to thank the Northern Ireland Department of the Economy for funding the related PhD scholarship and Dublin Institute of Technology for the use of the data set in this study, with particular thanks to DIT research student Kevin Tighe who played a central role in data collection and collation.

#### 6. REFERENCES

- [1] S. Lohr, "Big Data's Impact in the World - The New York Times," *The New-York Times*, 2012. [Online]. Available: <https://www.nytimes.com/2012/02/12/sunday-review/big-datas-impact-in-the-world.html>. [Accessed: 04-Apr-2018].
- [2] J. Wang, Y. Chen, S. Hao, X. Peng, and L. Hu, "Deep Learning for Sensor-based Activity Recognition: A Survey," pp. 1–10, 2017.
- [3] J. Poushter and R. Stewart, "Smartphone Ownership and Internet Usage Continues to Climb in Emerging Economies But advanced economies still have higher rates of technology use."
- [4] Ofcom, "Digital divide narrows, but 1.1m UK homes and businesses cannot get decent broadband - Ofcom," 2017. [Online]. Available: <https://www.ofcom.org.uk/about-ofcom/latest/media/media-releases/2017/connected-nations-digital-divide>. [Accessed: 04-Apr-2018].
- [5] Z. Zhang, L. Liparulo, M. Panella, X. Gu, and Q. Fang, "A Fuzzy Kernel Motion Classifier for Autonomous Stroke Rehabilitation," *IEEE J. Biomed. Heal. Informatics*, vol. 20, no. 3, pp. 893–901, 2016.