# A case study for human gesture recognition from poorly annotated data

Mathias Ciliberto, Daniel Roggen, Lin Wang, Ruediger Zillmer



**Published in**
UbiComp '18 Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers

**Link to external publisher version**
https://doi.org/10.1145/3267305.3267508

# A Case Study for Human Gesture Recognition from Poorly Annotated Data

**Mathias Ciliberto**
Wearable Technologies Lab
Sensor Technology Research Centre
University of Sussex, UK
m.ciliberto@sussex.ac.uk

**Daniel Roggen**
Wearable Technologies Lab
Sensor Technology Research Centre
University of Sussex, UK
daniel.roggen@ieee.org

**Lin Wang**
Wearable Technologies Lab
Sensor Technology Research Centre
University of Sussex, UK
w23@sussex.ac.uk

**Ruediger Zillmer**
Unilever R&D Port Sunlight
ruediger.zillmer@gmail.com



## Abstract

In this paper we present a case study on drinking gesture recognition from a dataset annotated by Experience Sampling (ES). The dataset contains 8825 "sensor events" and users reported 1808 "drink events" through experience sampling. We first show that the annotations obtained through ES do not reflect accurately true drinking events. We present then how we maximise the value of this dataset through two approaches aiming at improving the quality of the annotations post-hoc. First, we use template-matching (Warping Longest Common Subsequence) to spot a subset of events which are highly likely to be drinking gestures. We then propose an unsupervised approach which can perform drinking gesture recognition by combining K-Means clustering with WLCSS. Experimental results verify the effectiveness of the proposed method.

## Author Keywords

Gesture recognition; dataset annotation; activity discovery; dataset curation

## ACM Classification Keywords

H.3.m [Information storage and retrival]: Miscellaneous

## Introduction

Gesture recognition has applications in several fields such as healthcare and sports [7]. In order to create a reliable gesture recognition system, it is important to have a well-annotated dataset [1]. However, creating high-quality datasets may require to rely on lab-like naturalistic environments, with limited ecological validity [11]. Activity recognition research generally strives to employ datasets with unrealistically "perfect" ground truth annotations. In an ecologically valid data collection, however, it is likely that a highly valuable dataset is acquired, but that only poor quality annotations are available.

Experience sampling (ES) is a real-time annotation approach done by users themselves a mobile device [13]. This allows more ecologically valid data collection in everyday life (e.g. no need to video record the experiment). However, ES can lead to the following issues: i) the synchronisation between the activity performed and the label annotated by the user is generally of poor quality, with the user annotating the activity after the event, or combining multiple activities in a single annotation; ii) the user may forget to label an event, iii) the user may annotate an activity with the wrong label.

In this work, we investigate how to make sense of a dataset with high business value, which has been annotated through ES, which led to numerous deficiencies in the annotation quality. The dataset contains drinking gestures annotated by the users with a mobile application. The dataset was collected in an office environment using a 3-axis accelerometer and it is made by 8825 "sensor events", with 1808 "drink events" annotated by users through ES. Using this dataset, we aim to address two main challenges: i) to understand why the quality of the annotation is low and consequently how would it be possible to improve in future data collection and ii) to understand whether it is still possible to use such big dataset without relying on the annotations for spotting drinking gestures and how. The contributions of this work are:

- A study of the annotations. We analyse the user annotations, their distribution in time during the data collection, and their relation to the sensor events, in order to understand the causes of the low quality and where the data collection process can be improved.
- A template matching approach, based on Warping Longest Common Subsequence (WLCSS) [8], to extract a subset of drinking gestures, within a certain level of confidence. This subset will allow the dataset to be used for research purposes.
- An unsupervised algorithm (K-Means) adapted to template matching. This algorithm is a new variation of K-Means [4] where the WLCSS is used as distance measure. It allows to cluster gestures based on the raw signal of the sensors. At the same time, it clusters gestures taking in account the variation in the way they can be performed, by using WLCSS which has been successfully used for robust gesture detection [8].

## Related work

The quality of annotations obtained through ES can be poor [13]. Annotations issues can include time shift of a label with respect to the activity, as well as wrong or missing labels [9].

Some approaches suggested to improve ES with manual re-annotation [13]. This is not feasible economically for a large number of users and however, in [13] the quality of

the annotations was still not sufficient for the training of machine learning algorithms. The impact of ES on activity recognition has been studied in [2]. However, the authors simulated the ES in a controlled environment and they used only the data corresponding to the user annotations.

The problem of poorly labelled data can be tackled during the annotation itself or during the training of the machine learning algorithm. A method useful to reduce the effort of the users while annotating their activity has been proposed in [10]. The authors proposed a one-time point annotation method that requires the users to only label a single moment per activity rather than specifing the beginning and the end. The method then recognizes automatically the boundary of the activity in the annotated signal. Nevertheless, it requires that the labels are within the execution interval of the activity.

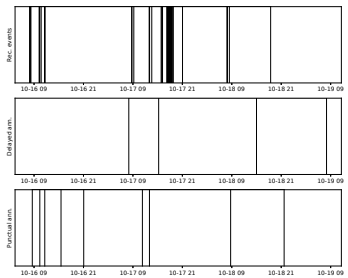
Unlabelled or poorly labelled data are available in big quantities nowadays due to the large diffusion of sensing devices, such as smartphones and wearables devices. For this reason, methods such as semi-supervised learning, active learning and unsupervised learning have been applied in order to extract useful information from sparsely annotated data. A combination of active learning and semi-supervised learning has been studied in [12]. The authors used a dataset of daily activities collected with two subjects wearing accelerometer sensors and motioned tracked with infrared sensors. This approach however uses a decision window of 30 seconds long and thus is not suitable for recognizing gestures that occur in a short time. Unsupervised learning has been successfully applied to activity recognition in [5] and more recently in [3]. In the latter, an activity discovery method based on clustering is proposed to help with ES, although it is designed for periodic movements rather than sporadic gesture. Unsupervised learning has been applied to gestures clustering in [14], where a K-Means clustering has been evaluated specifically for hand gestures.

Several studies have tried to address the challenge of activity recognition from poorly annotated data. While most of them used synthetic dataset and focused on periodic or long activity (such as walking, running, etc.), to the best of our knowledge none of them applies to drinking gestures collected in a real-life office environment.

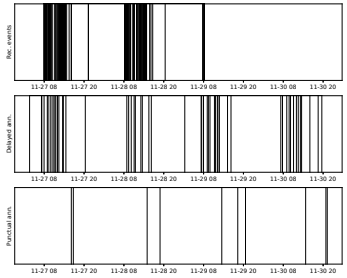
## Dataset

The dataset was collected by providing a set of mugs to 60 users (one mug per user) in an office environment. Each user collected data for a period of 4 days. Each mug was instrumented with a logger comprising a 3-axis accelerometer [15]. The loggers were placed in a hollow at the bottom of each mug. As the mugs were customly made, the positioning of the loggers was not the same in all mugs. The loggers sample acceleration at 20 Hz, with a timestamp in ms. In order to save power, they start logging acceleration when a movement is detected. After 5 seconds of inactivity they automatically stop the recording, without record the inactivity period. We use the term *sensor events* to refer to every recording performed by the loggers that lasts at least 4 seconds (as configured on the loggers for this data collection). Therefore, sensor events can occur for a variety of reasons: moving the mug on the desk, washing it, drinking from the cup, etc.

The data annotation was performed through experience sampling by the users themselves. They labelled each drinking event manually using an Android application installed on their smartphones. Each annotation could be *punctual* or *delayed*. An annotation is considered punctual when it was entered immediately after the drinking event.




**(a)** *User 109*




**(b)** *User 461*

**Figure 1:** Example of annotations of two different users, over the 4 days period. The start time of the sensor events are displayed in the top plot of each figure, one thin line per event. The delayed and punctual annotations inserted by the users are displayed respectively in the second and the third plot of both figures. The X-axis reports date and time, in the format "MM-DD HH". It is also possible to notice the differences in the way two users annotated the drinking events.

It is considered delayed, when it refers to an event in the past. The users could specify in the application whether their annotation was punctual or delayed. However they did not have to provide an indication as to how much the delay was. Furthermore, there were no guidance indicating after how much time an annotation should be considered "delayed" rather than "punctual".

The resulting dataset is made by 8825 sensor events, 1808 user annotations, of which 1477 marked as "punctual" and 331 as "delayed". The percentage of annotated gestures with respect to the total amount of sensor events is of 20.5%.

## User annotation analysis

We aim to analyse the causes of the poor annotations in order to improve future data collections, as well as helping during the next steps of this study.

Figure 1 indicates the main challenge of the annotation protocol, which is how users understood differently how and what to annotate. The data collection protocol did not require participants to annotate drinking solely when using the instrumented mugs: they could annotate drinking as well when using regular mugs. It might happen that users annotated drinking events performed using other cups. The protocol did also not specify what to consider as a "drinking event". Users could interpret it as referring to a single sip, multiple sips, or drinking the entire cup. It is also possible to notice how the annotations are not perfectly aligned with the sensor events.

We also studied the distribution of the labels, per user, over the 4 days of data collection. It could help to understand the users' commitment in annotating their drinking gestures, assuming they were keeping the same drinking habits among all the days. This may be useful in order to spot days for which the annotations can be more reliable. The results are presented in Figure 2. While there is no significant change between day 1 and day 2, with an average increase in the number of annotation of 0.24%, starting from day 3 the engagement decrease by 11% on average among all the users. The plot shows also a great variability in the data: there were users that increased their commitment over the 4 days, as well as users for which the commitment decreased over the 4 days.

From the analysis of the annotations, it can be concluded that they were not reliable enough to be used together with the data for the supervised classifier training.

## Gesture classification

In order to make the collected dataset useful for drinking gesture recognition, each event recorded by the sensors had to be classified in drinking/non-drinking. As highlighted previously, the users annotations cannot be used as-is as they are not accurate enough. A manual relabelling of the entire dataset was unfeasible given the lack of any video recordings.

We developed an approach based on a template matching method (TMM) to automatically spot a subset of events which are believed to be drinking gestures with a certain confidence value. The approach then uses few events which are manually identified as drinking events with high confidence to train the TMM.

*Data processing and training set selection*

We used a heuristic method to select a few sensor events as the training set. We performed a few drinking gestures using the same instrumented mug and discovered that the Z-axis of the accelerometer quite clearly indicates the gesture of lifting the cup to drink. A template of such gesture is displayed in Figure 3. A subset of gestures visually sim-

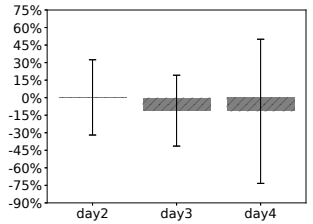



**Figure 2:** Change in the user commitment in the annotation during day 2, 3, and 4. The gray bars represents the percentage of annotations for each day with respect to day 1. Day 2 displays an increment of 0.24%; Day 3 and 4 an average decrease of 11% in the number of annotations. The vertical bars represents the standard deviation for each day.

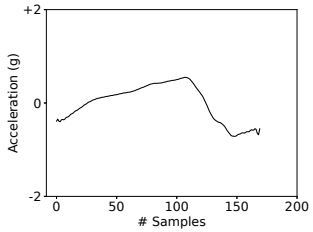
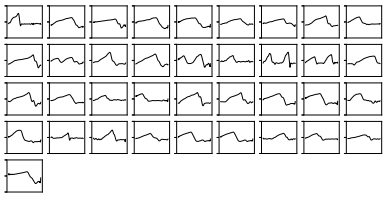
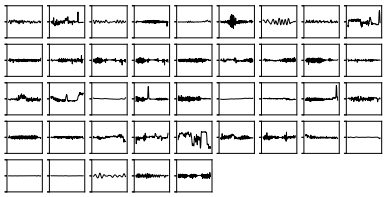



**Figure 3:** Template of a drinking gesture, performing a single sip.



**(a)** *Drinking gestures*



**(b)** *Non-drinking gestures*

**Figure 4:** Training set of gesture. 4a displays the templates chosen as drinking gestures. 4b shows those selected as no-drinking gestures. All the plots show the templates downsampled to the fixed length of 170 samples (X-axis). The Y-axis represents the acceleration within a range of $\pm$ 2g.

ilar to this template was selected manually from the entire set of the available gestures. We selected this subset trying to include some variability in the way the drinking gestures were performed. Another subset of non-drinking gestures was selected too, chosing the templates that were very visually different from the drinking gestures. The training set is displayed in Figure 4. It is formed by 78 events: 37 drinking gestures (4a) and 41 non-drinking gestures (4b).

All the instances in the dataset were filtered using a Butterworth low pass filter with cut off frequency set to 10 Hz. They were also resampled to a fixed number of samples. The number of samples was selected as the average length of a drinking gesture, which is 170. This step was performed in order to reduce the impact of non-drinking events that can last longer time than drinking gestures (e.g. washing the cup, moving the cup around the office, etc.).

*Template matching using WLCSS*

The Warping Longest Common Subsequence (WLCSS) [8] is an algorithm developed for template matching in real-time applications. Using dynamic programming, the algorithm can compute a matching score between a template and a stream, updating it at every new sample of the stream. It can be used for gesture recognition as it can handle gestures performed with variation in their speed of execution. This is achieved by three parameters: reward (R), penalty (P) and acceptance distance ($\epsilon$). The algorithm is shown in (1).

$$M(i,j) = \begin{cases} 0 & \text{if} \quad i \leq 0 \quad or \quad j \leq 0 \\ M(j-1,i-1) + R & \text{if} \quad |S(i) - T(j)| \leq \epsilon \\ max \begin{cases} M(j-1,i-1) - P \cdot |S(i) - T(j)| \\ M(j-1,i) - P \cdot |S(i) - T(j)| \\ M(j,i-1) - P \cdot |S(i) - T(j)| \end{cases} & \text{if} \quad |S(i) - T(j)| > \epsilon \end{cases} \tag{1}$$

The matching score M(i,j) is computed as function of the previous scores, by adding a reward (R) when the distance between the i-th stream sample ($S(i)$) and the j-th template sample ($T(j)$) is below an acceptance distance ($\epsilon$), or by subtracting a penalty (P) proportional to the distance, when this is above $\epsilon$. In addition to R,P, and $\epsilon$, WLCSS needs a threshold T. As WLCSS computes a matching score (M) between an instance in the dataset and a template, T is used to define whether an instance matches with the template (M$\geq$T) or not (M$<$T). The value of the threshold is related to the specificity and the sensitivity of the matching algorithm: a high threshold means high specificity, while a low

threshold means high sensitivity. The values of R,P,$\epsilon$ and T must be found during the training phase. We optimized this based on an evolutionary optimization technique.

*WLCSS optimization using evolutionary algorithm*
We optimise the values of R, P, e, T to maximise the ability of WLCSS to distinguish drink from non-drink using an evolutionary algorithm (EA). We used the EA in order to optimize the values of the parameters starting from a randomly generated population. Each individual of the population is an array containing the 4 parameters. The EA evolves this population through the usual selection, mutation and crossover operators [6]. Here, the F1 score is used for the selection. The optimization process stops after a predefined number of iterations, in this case .

*Confidence computation*
Given the unreliability of the labels in the dataset, it is not possible to evaluate precisely the correctness of a match for this particular dataset. For this reason, we opted to provide a confidence level for each gesture as output of our method rather than a simple match/no-match. The confidence level was assigned using Four Parameter Logistic Regression:

$$y = d + \frac{a - d}{1 + (\frac{M}{T})^b}$$

where $y$ is the confidence level, $M$ is the matching cost, and $T$ is the threshold. The function, displayed in Figure 5, provides a confidence value in the range [0:1]. The range is defined by the parameters $a = 0$ and $d = 1$. The parameter $b$, which define the slope of the function curve, was set manually to 5. Using a fixed interval for the confidence makes its value unrelated from the absolute value of $T$, which can vary according with the parameters $R$, $P$ and $\epsilon$.

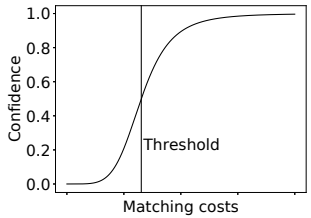



**Figure 5:** 4 Parameter Logistic Regression function used to assign a confidence with the respect to the threshold.

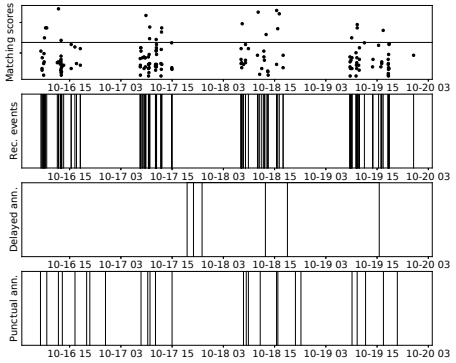



**Figure 6:** Comparison of WLCSS matching scores with recorded data for a single user. From the top: the WLCSS matching scores and the threshold T, as horizontal line (first plot), the start time of the sensor events (second plot), and the user annotations delayed and punctual (respectively third and fourth plot). The data are for 4 days period, with the X-axis reporting date and time in the format "MM-DD HH"

*Evaluation*
We trained the system using the EA and the subset of instances selected as training set. As the EA is a stochastic process, we repeated the training 10 times, and we picked the best values of R=68, P=0, $\epsilon$=28 and T=3364 for the WLCSS. With these values, we run the algorithm on the entire dataset, by using the template displayed in Figure 3, which was selected manually from the training set as template.

Figure 6 displays a comparison of the sensor events and the annotations for a single user, with the corresponding matching scores. In the figure, it is also reported the threshold: the matching costs $>= T$ are those which are detected as drinking gesture with a confidence $\geq$50%. The percentages of total detected gestures compared to the total number of events are displayed in Table 1, for different confer-

**Table 1:** Number of drinking gestures detected for some confidence levels. The percentages are with the respect to the total number of sensor events in the dataset (8825).

| Confidence | # gestures | % |
|---|---|---|
| $\geq$ 25% | 1481 | 17% |
| $\geq$ 50% | 942 | 10% |
| $\geq$ 75% | 543 | 6% |

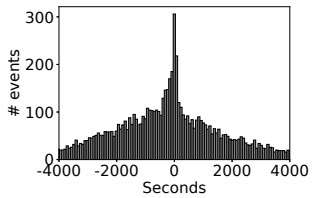



**Figure 7:** Cross-correlogram representing the distribution of the delays (in seconds) between the user annotations and all the events recorded by the loggers.

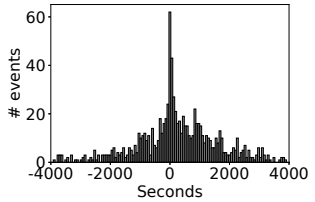



**Figure 8:** Cross-correlogram representing the distribution of the delays (in seconds) between the user annotations and the events detected with a confidence $\geq$50%.

ence values. The low percentages are due to the nature of the sensors, which were collecting all sort of movements such as moving the mug on the desk, washing it or even accidental movements. It is important to note that the number of detected events is also lower than the number of user annotations (1808). This may be a result of the data collection protocol which did not specify to annotated only the drinking movements performed with the instrumented mug.

We studied the relation between user annotations, sensor events and detected gestures. To do this, we assigned to every recorded event the closest user annotation in time. Then, computing the time difference between the sensor events and the corresponding closest annotations, we created the cross-correlogram displayed in Figure 7. Figure 8 presents the distribution of the same time differences, but considering only the gestures detected by WLCSS with a confidence $\geq$50%. The latter plot presents a more pointy distribution, confirming that WLCSS detected events that were actually closer in time to the user annotations.

## Unsupervised learning

We evaluated also an unsupervised approach in order to classify the gestures in drinking/non-drinking as it does not require a training set. We developed a custom method based on K-Means. We modified K-Means in order to make it able to cluster gestures performed with variation in their speed of execution.

*K-Means with WLCSS*

K-Means is a clustering technique that aims to partition $n$ observation in $k$ clusters. Each observation belongs to the cluster with the closest mean. It can be used for unsupervised learning by clustering the input data based on a distance measure. The algorithm is based on two steps, assignment step and update step, which are repeated until a stopping criteria is met [4]. This criteria can be reaching a maximum number of iterations, the change of the clusters in the update step in below a thresholds, etc. We implemented a modified version of the K-Means, where WLCSS was used a distance measure in place of the Euclidean distance. The assignment in our implementation is modified as following:

$$\arg\max_{c_i} WLCSS(x, c_i)$$

where $x$ is a sensor events, $c_i$ is the centroid for the i-th cluster. The function *argmin* is replaced by *argmax* as WLCSS compute a matching score rather than a distance. The update step is unmodified.

*Evaluation*

We compared our version of K-Means (named K-Means-WLCSS) against the standard version that uses the Euclidean distance in order to assign each instance to the closest cluster. We applied both the implementation on the training set from the previous step, with $k = 2$ as the goal was to distinguish between drinking and non-drinking gestures. As all the instances were resampled to the same length, they could be used as feature vectors for both the implementations, without dealing with different lengths of the feature vectors (in this case the resampled raw signal). Applying the algorithms on the training set allowed us to compare the clustering results with the labels assigned manually to each gesture, during the data selection. Figure 9 presents a visual comparison between the clusters obtained with the two versions of K-Means. For both the implementations, Cluster 1 seems to include mainly the drinking gestures, while Cluster 2 the non-drinking gestures. We used this consideration in order to evaluate the performance of the two algorithms computing precision, recall and F1 score, presented in Table 2. They are computed by comparing the clustering of K-Means with the manual labels

**Table 2:** Precision, Recall and F1 score for K-Means and K-MeansWLCSS computed on the training set. They are computed using the manual labels assigned to each instance in the training set as ground-truth. The majority of the gestures in a cluster is used as classification label for the K-Means implementations.

| | K-M | K-M_WLCSS |
|---|---|---|
| Precision | 100% | 92.11% |
| Recall | 62.16% | 94.6% |
| F1 | 77% | 93% |

of the instances in the extracted subset. K-Means_WLCSS increased the F1 score by 16%, being able to detect more variations in the drinking gestures as it is also visible from Figure 9. It was able to cluster correctly drinking gestures composed by two sips, such as the instances 8, 16, and 21 of Figure 9c.

## Discussion

We discovered that the main issue for this dataset was the data collection protocol which was too relaxed. More precise instructions would increase considerably the quality of the data. Simultaneously, asking the users for more precision in following the protocol should be balanced with shorter sessions of data collection, as we noticed how the user commitment decreases over 4 days of continuous data collection. Lastly, as it has been demonstrated that experience sampling is not reliable, we recommend to increase the effort in the setup of the experiment by including a video recording. It would dramatically increase the quality and re-usability of the dataset, although it would require additional time for the labelling of the data. In order to reduce this effort, the video recordings can be used to precisely annotate just a small portion of the entire dataset. This well-annotated subset, which can be also collected a-posteriori, or can be used as training set instead of extracting one through heuristic. The well-annotated dataset would also allow to evaluate more precisely the performance of the TMM or any other classifier on the complete dataset, through statistical analysis.

In this study we extracted a subset of events which can be considered as drinking gestures with a certain confidence. This extracted subset can be potentially used to re-train the TMM for a more reliable gesture recognition system. The re-training phase can be performed using gestures with different levels of confidence: an higher level of confidence would increase the specificity of the found gestures. Decreasing this value would increase the sensitivity, potentially including more variations of the drinking gestures.

In our approaches, we used only the accelerometer signal recorded on the Z-axis. Using this axis, the lifting gesture was clear (see Figure 3), but it does not necessarily mean that the user lifts the cup and drinks. A more extensive and potentially robust approach would include also the X and the Y axis, computing the combined acceleration (on X and Y), in order to spot the actual sipping gesture.

Finally, we aimed to evaluate how an unsupervised learning technique can be used in order to extract drinking gestures from a poorly labelled dataset. We implemented a modified version of K-Means which uses WLCSS as distance measure for the assignment step. The results are promising: with 2 clusters it managed to differentiate between drinking and non-drinking gesture with an 93% F1-score, although on a limited number of sensor events. A more extensive evaluation can be performed on the entire dataset, although without a reliable ground-truth, a validation of the results, in this case, could be difficult.

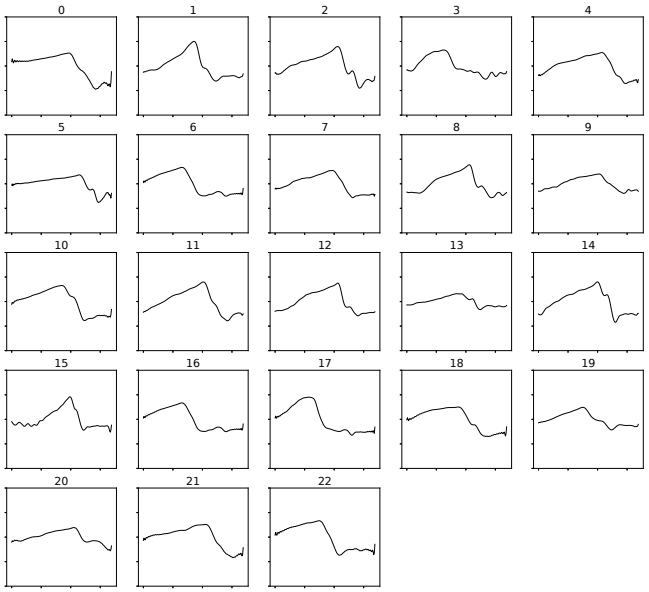
## Conclusion

In this study, we investigate how to extract knowledge from a poorly labelled dataset of drinking gestures. We analysed the user annotations in order to get qualitative information on how to improve the data collection. We exposed how the loose protocol created most of the problems and we highlighted the need of providing more precise instructions to the users. Then, by selecting instances manually and using a template matching algorithm, we demonstrated that it is possible to extract a subset of instances which are actually drinking gestures within a certain level of confidence. We proved that an unsupervised approach based on K-Means

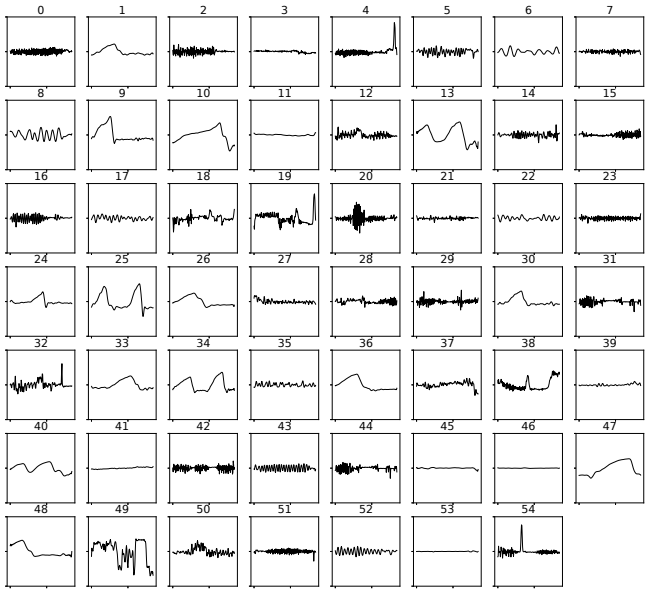
and WLCSS can improve the clustering of gesture over the standard K-Means implementation. Our method outperformed the baseline method by including a wider variety of drinking gestures and increasing F1-score by 16%.

## REFERENCES

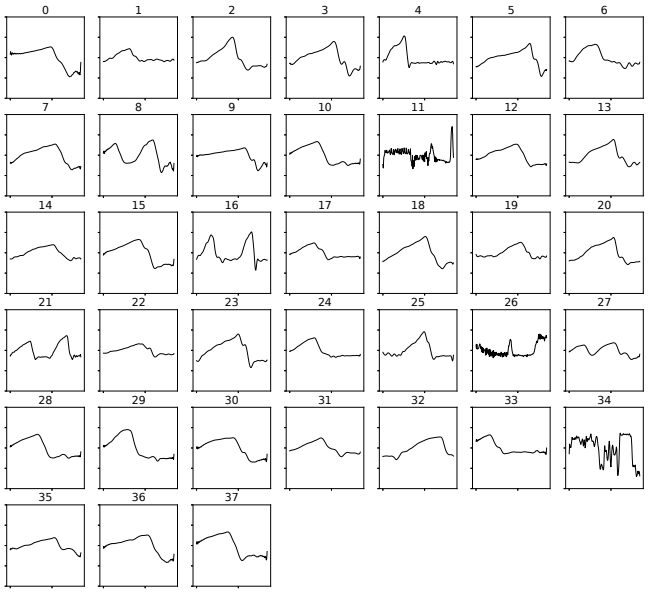

1. A. Bulling et al. A tutorial on human activity recognition using body-worn inertial sensors. *ACM Computing Surveys*, 46(3):1–33, 2014.
2. W. Duffy et al. Addressing the problem of Activity Recognition with Experience Sampling and Weak Learning. *Proceedings of SAI Intelligent Systems Conference*, pages 1–6, 2018.
3. H. Gjoreski and D. Roggen. Unsupervised online activity discovery using temporal behaviour assumption. *Proceedings of the ACM International Symposium on Wearable Computers*, pages 42–49, 2017.
4. J. A. Hartigan and M. A. Wong. Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108, 1979.
5. Y. Kwon et al. Unsupervised learning for human activity recognition using smartphone sensors. *Expert Systems with Applications*, 41(14):6067–6074, 2014.
6. Z. Michalewicz. Evolution strategies and other methods. In *Genetic algorithms+ data structures= evolution programs*, pages 159–177. Springer Berlin Heidelberg, 1996.
7. S. Mitra et al. Gesture recognition: A survey. *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, 37(3):311–324, 2007.
8. L.-V. Nguyen-Dinh et al. Improving online gesture recognition with template matching methods in accelerometer data. *International Conference on Intelligent Systems Design and Applications*, pages 831–836, 2012.
9. L.-V. Nguyen-Dinh et al. Robust Online Gesture Recognition with Crowdsourced Annotations. *Journal of Machine Learning Research*, 15:3187–3220, 2014.
10. L.-V. Nguyen-Dinh et al. Supporting One-Time Point Annotations for Gesture Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(11):2270–2283, 2017.
11. D. Roggen et al. Collecting complex activity datasets in highly rich networked sensor environments. *Proceedings of International Conference on Networked Sensing Systems*, pages 233–240, 2010.
12. M. Stikic et al. Exploring semi-supervised and active learning for activity recognition. *IEEE International Symposium on Wearable Computers*, pages 81–88, 2008.
13. M. Stikic et al. Activity Recognition from Sparsely Labeled Data Using Multi-Instance Learning. In *International Symposium on Location- and Context-Awareness*, pages 156–173. IEEE, 2009.
14. X. Zhang et al. A framework for hand gesture recognition based on accelerometer and EMG sensors. *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans*, 41(6):1064–1076, 2011.
15. R. Zillmer et al. A robust device for large-scale monitoring of bar soap usage in free-living conditions. *Personal and Ubiquitous Computing*, 18(8):2057–2064, 2014.



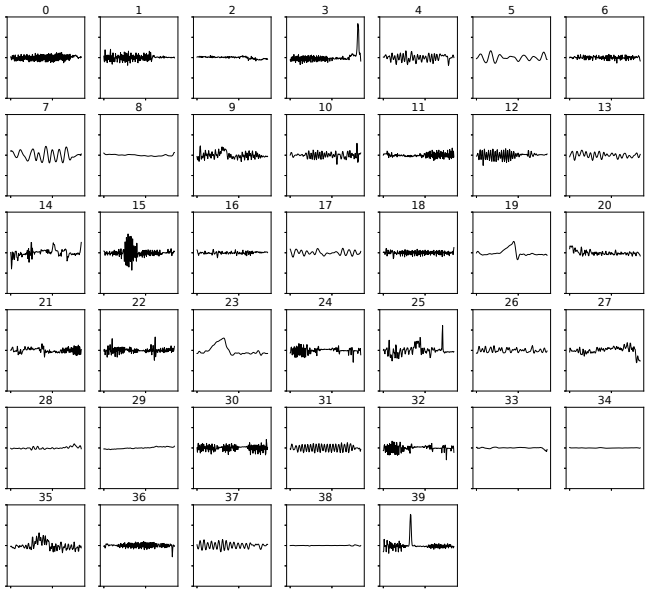
**(a)** *Cluster 1 - K-Means*



**(b)** *Cluster 2 - K-Means*



**(c)** *Cluster 1 - K-MeansWLCSS*



**(d)** *Cluster 2 - K-MeansWLCSS*

**Figure 9:** Comparison between clusters obtained with K-Means using Euclidean distance (top left, top right), and using WLCSS (bottom left, bottom right).