



ALMA MATER STUDIORUM  
UNIVERSITÀ DI BOLOGNA

## ARCHIVIO ISTITUZIONALE DELLA RICERCA

### Alma Mater Studiorum Università di Bologna Archivio istituzionale della ricerca

ActivityExplorer: A semi-supervised approach to discover unknown activity classes in HAR systems

This is the final peer-reviewed author's accepted manuscript (postprint) of the following publication:

*Published Version:*

ActivityExplorer: A semi-supervised approach to discover unknown activity classes in HAR systems / Brighi M.; Franco A.; Maio D.. - In: PATTERN RECOGNITION LETTERS. - ISSN 0167-8655. - ELETTRONICO. - 151:(2021), pp. 340-347. [10.1016/j.patrec.2021.08.029]

*Availability:*

This version is available at: <https://hdl.handle.net/11585/865972> since: 2022-02-24

*Published:*

DOI: <http://doi.org/10.1016/j.patrec.2021.08.029>

*Terms of use:*

Some rights reserved. The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. For all terms of use and more information see the publisher's website.

This item was downloaded from IRIS Università di Bologna (<https://cris.unibo.it/>).  
When citing, please refer to the published version.

(Article begins on next page)

This is the final peer-reviewed accepted manuscript of:

**Marco Brighi, Annalisa Franco, Dario Maio, ActivityExplorer: A semi-supervised approach to discover unknown activity classes in HAR systems, Pattern Recognition Letters, Volume 151, 2021, Pages 340-347**

The final published version is available online at:  
<https://dx.doi.org/10.1016/j.patrec.2021.08.029>

Terms of use:

Some rights reserved. The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. For all terms of use and more information see the publisher's website.

*This item was downloaded from IRIS Università di Bologna (<https://cris.unibo.it/>)*

***When citing, please refer to the published version.***

### **Research Highlights (Required)**

- A semi-supervised activity recognition approach able to identify unknown activity classes.
- Approach designed to deal with small-sample set scenarios with limited amount of training data.
- Approach based on affinity propagation clustering able to automatically identify the number of clusters.
- Results on public datasets confirm the efficacy of the proposed approach.



# ActivityExplorer: a semi-supervised approach to discover unknown activity classes in HAR systems

Marco **Brighi**<sup>a</sup>, Annalisa **Franco**<sup>a,\*\*</sup>, Dario **Maio**<sup>a</sup>

<sup>a</sup>*Department of Computer Science and Engineering, University of Bologna, Via dell'Università 50, Cesena 47521, Italy*

## ABSTRACT

A semi-supervised activity recognition system is here proposed to deal with partially labeled video-sequences, where the uncertainty in the data comes from two different factors: only a subset of the data has a class label assigned and only part of the activity classes are known. In particular, the paper presents ActivityExplorer, an approach able to identify clusters of similar activity patterns within the dataset and to identify those clusters that might correspond to new activity classes, still unknown to the recognition system. These capabilities are realized thanks to a combination of metric learning, used to determine a suitable subspace for pattern classification, an advanced clustering technique and ad hoc indicators defined to estimate the membership of each pattern to known classes and possibly identify new activities.

© 2024 Elsevier Ltd. All rights reserved.

## 1. Introduction and related works

Human action recognition (HAR) is a topical issue in the development of vision systems, due to its numerous applications such as video-surveillance, robotics, instructional video analysis (Tang et al. (2020a)) or ambient assisted living, just to mention a few. HAR recognition has been widely studied in the past years and a variety of techniques for activity detection or recognition have been proposed in the literature. Interested readers can refer to the recent survey by Minh Dang et al. (2020) for an overview of the state-of-the-art. This work is motivated by the observation that many works in the literature deal with a sort of "closed-set" scenario, where all the activities of interest are known to the system whose only task is therefore limited to assign one of the known class labels to newly incoming data. This assumption is of course a great limitation to the potentialities of HAR systems which would gain a much more significant role if they were able to autonomously identify also unknown but repeatedly observed activity patterns. Approaches of this kind are usually defined semi-supervised, since they are able to deal with a partially labeled dataset of activity patterns in the hypothesis that only a subset of the existing activities is known to

the system. This assumption is much more feasible in real environments where manually labeling data is very expensive, tedious and time consuming. Whereas the availability of labeled data is very limited, a huge amount of unlabeled data can easily be acquired by a hypothetical continuous monitoring system. Such unlabeled data usually neglected by most approaches, can be successfully by continuous learning strategies, such as incremental template updating approaches (Franco et al. (2020b)), aimed at improving the known activities representation, or unknown activity discovery techniques; the authors of Chen et al. (2020) show another interesting use of unlabeled data as auxiliary data exploited to better deal with spatio-temporal variations.

The problem of unsupervised and semi-supervised activity recognition has been addressed by some works in the literature. Driven by the increased popularity and widespread diffusion of wearable devices with different sensing capabilities, most of the relevant approaches are based on the analysis of signals obtained through low-level sensors which provide simpler and easily manageable data. Smartphones are certainly the most studied devices. In Lu et al. (2017) accelerometer data are encoded by nineteen features used to construct a graph-based representation; clustering is then proposed for activity discovery. Accelerometer and gyroscope data are used by the authors of Kwon et al. (2014) who combine features from time and frequency domains to represent the signals acquired by smart-

<sup>\*\*</sup>Corresponding author: Tel.: +39-0547-338847;  
e-mail: [annalisa.franco@unibo.it](mailto:annalisa.franco@unibo.it) (Annalisa Franco)

phones; in Li et al. (2014) different approaches are compared for unsupervised feature learning from accelerometer and gyroscope data. In this case Gaussian mixture models and clustering are used for unsupervised learning. Other sensing devices have also been explored, such as wearable wrist bands in Bai et al. (2019), inertial ring and bracelet in Moschetti et al. (2017) or other inertial sensors at different body parts in Trabelsi et al. (2013). A combination of different environmental sensors, as well as information about interaction with objects are exploited by Riboni et al. (2016) to derive semantic correlations among activities and sensor events. Our work, however, focuses on vision-based approaches for activity recognition (Beddiar et al. (2020)) where the explicit interaction of users with acquisition devices is not necessary. Recently, some approaches based on RGB video sequences and deep learning techniques have been proposed to deal with large scale open-set activity recognition (Gutoski et al. (2020) and Yu et al. (2020)), action prediction from incomplete sequences Chen et al. (2021) or group activity recognition (Tang et al. (2019) and Tang et al. (2020b)) with interesting results. However, the adoption of deep learning techniques requires a huge amount of training data, not always available in a small-scale problem, an home environment for instance, where few users and few activity samples per user are generally available. In this scenario, we believe that also “traditional” computer vision techniques can achieve good results and real time processing capabilities even with limited computational power.

Many other approaches for activity recognition exploit RGB-D sensors (see for instance Wang et al. (2018) and Jaeyong Sung et al. (2012)) able to capture multiple data streams (RGB, depth images, skeleton data), thus enabling the development of multi-modal approaches (Ehatisham-Ul-Haq et al. (2019), Ihi-anle et al. (2020), Franco et al. (2020a)). In this context, a few works deal with unsupervised learning. Ong et al. (2013) exploit skeleton data extracted by RGB-D sensors for human activity detection using a K-means clustering approach. In Fernando et al. (2017) an approach aimed at matching the same activity sequence in different videos is provided, with the peculiarity of not exploiting supervision to identify such video segments. The approach is based on an unsupervised temporal encoding method and exploits the temporal consistency in human actions. An approach based on skeleton is proposed in Su et al. (2020) where an encoder-decoder recurrent neural network is used to cluster similar movements into the same cluster and distinct movements into distant clusters.

The main focus of this paper is on semi-supervised activity recognition in uncertain conditions where only a subset of the data is labeled and only some of the activity classes are known. To further increase the complexity of this task, we deal with vision-based approaches based on a complex activity encoding, often derived from multimodal data sources, generally resulting in high-dimensional feature vectors. Operating in sparse high-dimensional spaces is known to be very hard and the paper proposes an approach to address these issues by adopting some key components: i) a dimensionality reduction technique based on metric learning, more effective for pattern classification compared to the standard Euclidean distance, to make the data more

manageable and improving at the same time their discriminability; ii) a robust clustering algorithm able to autonomously identify the natural distribution of patterns into classes; iii) a mechanism able to “recognize” new activities as data clusters not adequately represented by the known activity classes. A general approach has been designed here without any specific assumption on the feature set used for activity encoding; the experimental results will demonstrate the effectiveness of our proposal on different datasets.

## 2. The proposed approach

The aim of this work is to describe a general approach able to discover new activities within a continuous data stream acquired in a typical home environment, in a simple and efficient way. The constant research and discovery of new activities carried out by users should allow to overcome the “closed” scenario, in which the activities are fixed and not modifiable, moving towards a more realistic scenario. Figure 1 draws a general outline of the proposal. Let  $TR = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_n, y_n)\}$  be a small initial set of data, available for training the activity recognition system; it consists of a set of  $n$  patterns where  $\mathbf{x}_i$  is a  $p$ -dimensional feature vector representing an activity pattern and  $y_i$  is the associated activity label (with  $y_i \in \{1, \dots, m\}$ ). The features used for activity encoding are strictly related to the specific dataset used and details on this aspect will be provided along with the dataset description in Section 3.1. A classifier is trained using the set of labeled data  $TR$  (see section 2.1); it will thus try to learn the function  $f(\mathbf{x}_i) = y_i$  able to associate each single feature vector  $\mathbf{x}_i$  to the activity that it represents  $y_j$ .

After the initial learning stage, the activity recognition system will have to classify incoming sequences under the hypothesis that also activities belonging to unknown classes are presented to the system. We thus assume that the set of incoming activities is  $Z = \{(\mathbf{z}_1, y_1), (\mathbf{z}_2, y_2), \dots, (\mathbf{z}_n, y_n)\}$ , with  $\mathbf{z}_i \in R^p$  and  $y_i \in \{1, \dots, m + s\}$  where  $m$  of the activity classes in  $Z$  are known to the activity classification system and the remaining  $s$  are unknown. Those  $s$  unknown classes represent exactly what the algorithm aims to discover.

The training and testing stages of the proposed approach, outlined in Figure 1, are described in detail in the next sections.

### 2.1. Training stage

This stage (see Figure 1a) consists of training a classifier on the set of  $m$  initially known activities. In order to maximize the effectiveness of the classifier and to reduce at the same time the dimensionality of the feature vectors representative of the activity sequences, a metric learning approach for dimensionality reduction is applied. Each feature vector  $\mathbf{x}_i \in R^p$  is therefore projected into a space  $R^q$ , where  $q \ll p$ . The literature on metric learning is huge and recently some interesting deep metric learning techniques have been proposed (see Kaya and Bilge (2019), Zheng et al. (2020), Song et al. (2016)), but they are unfeasible for the small-scale scenario analyzed in this work.

In this scenario, dimensionality reduction is aimed at discriminating the different activity classes, i.e. creating clusters of high density data belonging to a single class. In Gold-

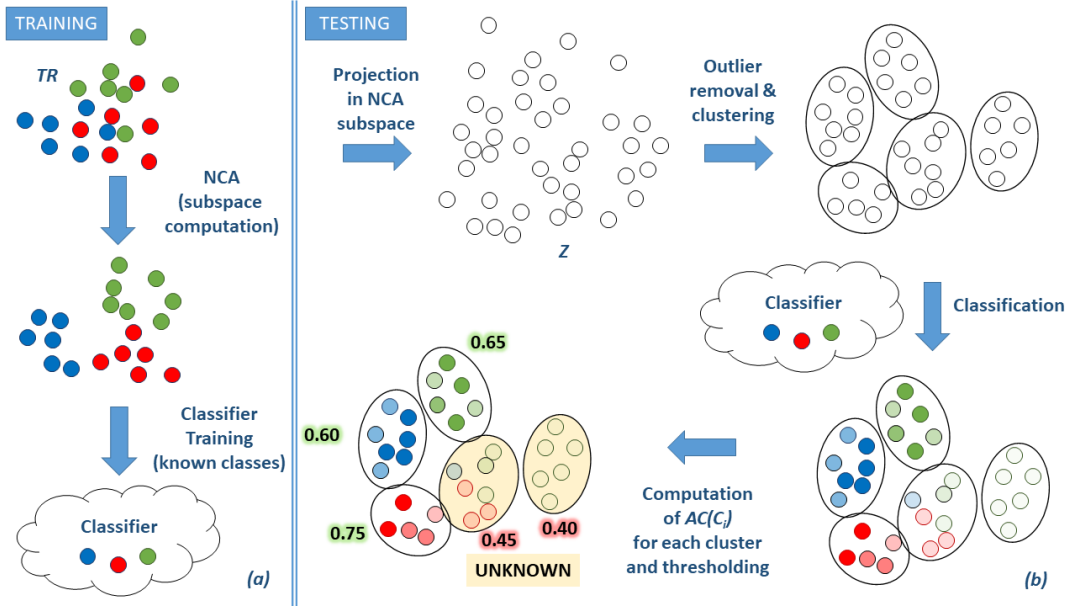


Figure 1: Outline of the training (a) and testing (b) stages of the proposed approach.

berger et al. (2005) an algorithm called Neighborhood Component Analysis (NCA), capable of responding to these requirements, is described. It is based on a metric learning technique and learns a linear transformation in a supervised fashion to improve the classification accuracy of a stochastic nearest neighbors rule in the transformed space. The goal of NCA is to learn an optimal linear transformation matrix  $A$  such that the average leave-one-out (LOO) classification performance is maximized in the transformed space. NCA tries to predict the class label of a single data point by consensus of its  $k$ -nearest neighbours, using a given distance metric. The set of nearest-neighbours of a generic vector  $\mathbf{x}_i$  can be quite different after the transformation in order to maximise the number of correct classifications. Unfortunately, it's hard to identify the optimal matrix as any objective function based on neighborhood points would be not differentiable. In particular, the set of neighbors for a point may undergo discrete changes in response to regular changes in the elements of  $A$ . NCA overcomes this difficulty by adopting an approach based on stochastic gradient descent. The entire transformed data set is considered as stochastic nearest neighbours using a softmax function of the squared Euclidean distance between a point and each other point in the space. In order to find the best dimensionality, we made several experiment with several values. After dimensionality reduction, the patterns in the training set  $TR$  are used to train a classifier.

## 2.2. Testing stage

The testing stage is outlined in Figure 1b. This is where ActivityExplorer acts to identify classes of unknown activities. To this aim, the patterns in the set of incoming activity sequences  $Z$  are first projected into the reduced feature space identified by NCA in the training stage; then, the following steps are executed:

- *Outlier detection.* In order to keep only the most representative samples, outliers are removed on the basis of the

Local Outlier Factor technique (see section 2.2.1).

- *Clustering.* The application of a clustering algorithm (see section 2.2.2) to this set of well-separated patterns allows to identify clusters related to known activities and to use subsequently the cluster information to identify unknown activity classes.
- *Activity discovery.* The patterns in each cluster are classified by the classifier trained in the learning stage and a score proportional to the probability of belonging to the known classes is assigned to each cluster (see section 2.2.3). According to this score, those groups that present very low membership values can be considered as representatives of new activity classes.

### 2.2.1. Outlier Detection

The step aims to improve the overall quality of the data by identifying and removing outliers, i.e. elements that differ significantly from the rest of the data. Their presence could strongly influence the subsequent steps and interfere with the identification of correct clusters. A variety of outlier detection algorithms is available in the literature. Since in this case some of the data classes are unknown, an unsupervised technique is needed. Local Outlier Factor (LOF) is an algorithm described in Breunig et al. (2000) for finding anomalous data points by measuring the local deviation of a given data point with respect to its neighbours. It works locally, since the anomaly score depends on how isolated the object is with respect to the surrounding neighborhood. Applying this type of algorithm in the reduced space means removing all the data points distant from the clusters representing an activity. The size of the neighborhood, in terms of  $k$ -neighbors, represents of course the fundamental parameter for the whole algorithm. Let  $d(\mathbf{x}_i, \mathbf{x}_j)$  be the Euclidean distance between  $\mathbf{x}_i$  and  $\mathbf{x}_j$  and  $k\_distance(\mathbf{x}_i)$  be the Euclidean distance of a generic point  $\mathbf{x}_i$  to its  $k$ -th nearest

neighbor. This distance measurement is essential to calculate the *reachability-distance* ( $RD$ ) between  $\mathbf{x}_i$  and another generic point  $\mathbf{x}_j$ :

$$RD(\mathbf{x}_i, \mathbf{x}_j) = \max\{k\_distance(\mathbf{x}_j), d(\mathbf{x}_i, \mathbf{x}_j)\} \quad (1)$$

The  $RD$  from  $\mathbf{x}_i$  to  $\mathbf{x}_j$  is the real distance between the two points only if this is greater than the  $k\_distance$  of  $\mathbf{x}_j$ . All the elements belonging to the  $k$ -nearest neighbors of  $\mathbf{x}_j$  are therefore considered to be equally distant and, in general, the reachability-distance is not symmetric. The set of points whose distance is less than the  $k\_distance(\mathbf{x}_i)$  is defined as  $N_k(\mathbf{x}_i)$ . To estimate if a point belongs to a local dense region, it is necessary to calculate the *local reachability density* ( $LRD$ ), which measures the average reachability of a point from its  $k$  neighbors:

$$LRD(\mathbf{x}_i) = \frac{|N_k(\mathbf{x}_i)|}{\sum_{\mathbf{x}_j \in N_k(\mathbf{x}_i)} RD(\mathbf{x}_i, \mathbf{x}_j)} \quad (2)$$

It is important to note that the reachability value does not concern the neighborhood starting from  $\mathbf{x}_i$ , but the exact opposite path. Given a point, the *local reachability density* is then compared with those of the neighbors:

$$LOF(\mathbf{x}_i) = \frac{\sum_{\mathbf{r} \in N_k(\mathbf{x}_i)} LRD(\mathbf{r})}{|N_k(\mathbf{x}_i)| \cdot LRD(\mathbf{x}_i)} \quad (3)$$

determining the *local outlier factor* ( $LOF$ ) value assigned to each point. If this value is close to one, this means that the element is comparable with those present in its neighborhood. A value greater than one denotes instead a point laying in a region with a lower density than that presents in its neighborhood. That element can therefore be defined as an outlier and can be removed from the dataset.

### 2.2.2. Clustering

After outlier removal, the data should be adequately grouped and each of these groups can therefore be captured by a clustering algorithm. Given a set of observations  $X = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ , a clustering algorithm should partition the data into a number of subsets containing elements that can be considered similar. In our work, algorithms requiring an a priori definition of the number of clusters are unfeasible since not all the activities (classes) in the data are known. Moreover, it has been observed that the density-based solutions encounter important difficulties. The application of metric learning on known activities only tends to separate groups of unknown activities, thus limiting the effectiveness of density-based techniques. For this reason we decided to adopt the Affinity Propagation approach proposed in Frey and Dueck (2007) and based on the concept of "message passing" between data points. The number of clusters is determined by the algorithm itself based on the characteristics of the data. In Affinity Propagation the data points can be seen as a network where all the data points send messages to others. This exchange leads to the identification of exemplars, which are the main representative points of their clusters. The algorithm iterates until it converges or it reaches a maximum number of iterations. During each iteration, two types of messages are exchanged. Let  $\mathbf{x}_i$  and  $\mathbf{x}_j$  be two different points of the dataset:

- the responsibility message  $r(\mathbf{x}_i, \mathbf{x}_j)$  contains a value that indicates how well-suited point  $\mathbf{x}_j$  is to serve as the exemplar for  $\mathbf{x}_i$ , considering all the other candidates exemplars;
- the availability message  $a(\mathbf{x}_i, \mathbf{x}_j)$  contains a value that represents how appropriate it would be for  $\mathbf{x}_i$  to pick  $\mathbf{x}_j$  as its exemplar, taking into account other points' preference for  $\mathbf{x}_j$  as an exemplar.

The similarity between pair of points is calculated as the negative squared distance (Xu and Wunsch (2005)):

$$s(\mathbf{x}_i, \mathbf{x}_j) = - \|\mathbf{x}_i - \mathbf{x}_j\|^2 \quad (4)$$

For each pair of points, the values of availability and responsibility are stored in two different matrices. Initially, all availabilities are generally set to zero and all responsibilities are set to the similarity between each pair of points. At each iteration, the values are updated:

$$r(\mathbf{x}_i, \mathbf{x}_j) = s(\mathbf{x}_i, \mathbf{x}_j) - \max_{j' \neq j} \{a(\mathbf{x}_i, \mathbf{x}_{j'}) + s(\mathbf{x}_i, \mathbf{x}_{j'})\} \quad (5)$$

$$a(\mathbf{x}_i, \mathbf{x}_j) = \begin{cases} \min\{0, r(\mathbf{x}_j, \mathbf{x}_i) + \sum_{i' \notin \{i, j\}} \max\{0, r(\mathbf{x}_{i'}, \mathbf{x}_j)\}\}, & i \neq j \\ \sum_{i' \neq j} \max\{0, r(\mathbf{x}_{i'}, \mathbf{x}_j)\}, & i = j \end{cases} \quad (6)$$

As the iterations proceed, the results tend to change until convergence or a maximum limit of iterations is reached. Among all the available points, the exemplars emerge, i.e. those who guide the aggregation of a set of points to create a cluster. In fact, for each points it is possible to identify its exemplar as the point that maximizes the sum of availability and responsibility:

$$exemplar(\mathbf{x}_i, \mathbf{x}_j) = \sum_{i' \neq i} \max\{a(\mathbf{x}_{i'}, \mathbf{x}_j) + r(\mathbf{x}_{i'}, \mathbf{x}_j)\} \quad (7)$$

Since the responsibility and availability do not exhibit symmetric properties, the exemplar value is consequently not symmetrical. Points that share the same exemplar constitute a single cluster, therefore, the number of exemplars determines the number of clusters.

### 2.2.3. Adherence to classes

At the end of the clustering stage, the patterns will be organized into a set of groups containing similar data. Of course, since the dataset is only partially labeled, it is reasonable to assume that some of these clusters can be linked to known activity classes, while others are likely to contain patterns of unknown classes.

To identify such clusters, the probabilistic classifier trained on the set of known activities during the learning stage is used (see section 2.1). Each pattern  $\mathbf{x}_i \in Z$  will be then classified by the classifier; let  $\hat{y}_i$  be the most probable class for  $\mathbf{x}_i$  and  $p(\hat{y}_i)$  the probability estimated by the classifier. In general, the patterns belonging to unknown activity classes will achieve quite low probability values and the clusters containing patterns associated with low probability values are likely to represent unknown activities. For each cluster of data  $C_i = \{\mathbf{x}_{i1}, \mathbf{x}_{i2}, \dots, \mathbf{x}_{il}\}$ , a score representing the overall cluster *adherence to known classes* is

computed by simply averaging the probabilities computed for each point in the cluster:

$$AC(C_i) = \sum_{\mathbf{x}_j \in C_i} p(\hat{y}_j) / |C_i| \quad (8)$$

On the one hand this value summarizes to what extent the elements of the cluster are likely to belong to known classes, on the other hand it indirectly allows to identify which clusters are most likely to refer to unknown classes, and therefore unknown activities. The clusters associated to an  $AC(C_i)$  value lower than a fixed threshold  $t$  will be identified as unknown and suggested to the user for a possible model updating.

### 3. Experiments and results

#### 3.1. Datasets and features

Several experiments have been carried out for performance evaluation on two publicly available datasets.

**Office Activity dataset (OAD)** The OAD dataset (Franco et al. (2017)) was acquired in our laboratory<sup>1</sup>. It includes video sequences of 14 activities performed twice by 20 subjects in a different environment from several perspectives. Each video is described by a feature vector composed of two main parts extracted from the use of an RGB-D camera. The first part of the vector represents RGB information obtained with Improved Dense Trajectories (Wang and Schmid (2013)), well-known for their excellent performance in action recognition tasks. Please refer to Franco et al. (2020b) for further details. The second part of the vector contains information based on skeleton joint positions and orientations, proposed in Franco et al. (2017). In conclusion, each video sequence is represented by a feature vector containing RGB (2000) and skeleton (100) information.

**KTH dataset** KTH is a well known video collection of human activities provided by Schudt et al. (2004). It includes 600 video sequences of 6 human activities performed several times by 25 subjects in four different scenarios. For our experiments, each video was encoded using the Dense Histogram of Optical Flow (HOF) feature (Perš et al. (2010)). For each pair of consecutive frames, the displacement of each pixel is identified by calculating the optical flow. The result is then divided into non-overlapping blocks and a histogram of orientations is stored for each of them. The descriptor of each pair of frames is given by the concatenation of these histograms. The final descriptor of each video is finally made using a Bag of Word model (BoW) (Wang et al. (2009)). The choice of  $K$ , i.e. the number of words that make up the dictionary for BoW computation, has been fixed to Internal tests allowed to identify the appropriate trade-off in the value 700, which also represents the size of the vector associated with each video of the dataset.

##### 3.1.1. Evaluation protocol

The efficacy of the proposed approach is tested, for each dataset, considering different combinations of known and unknown activity classes. Each configuration is repeated 5 times

by randomly sampling the activities from the dataset, and the average results are reported. The *training set* of known activities, needed to apply NCA dimensionality reduction and train a classifier, is randomly extracted by taking 30% of the samples available for each known activity. The remaining samples (70%) are included in the *test set*. The latter also contains samples of unknown activities in equal measure.

##### 3.1.2. Performance indicators

The efficacy of the proposed approach is evaluated using different indicators. For performance evaluation it is first necessary to determine a ground truth, i.e. to assign a class label  $y_{C_i}$  to each data cluster  $C_i = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ , simply identifying the most frequent class. Some clusters will be associated to known activity classes, other to unknown ones. Let's  $K = \{C_i | y_{C_i} \in \{1, \dots, m\}\}$  be the set of clusters associated to known activities and  $U = \{C_i | y_{C_i} \in \{m+1, \dots, m+h\}\}$  be the set of clusters relate to unknown activities. Of course, multiple clusters can be assigned to the same activity, and this is correct if we consider the natural variability in the execution of actions by different users. Based on the classes assigned as ground truth, different indicators are computed to evaluate the capability of correctly identifying unknown activity clusters (to be suggested to the user for possible updating). As illustrated in section 2.2.3, the clusters are sorted by the classes adherence indicator  $AC(C_i)$ . The clusters with low values are the less representative of known classes and are likely to be associated with one of the unknown activities. The algorithm can therefore fix a threshold  $t$  and label as unknown activities ( $\hat{U}$ ) all the clusters  $C_i$  whose  $AC(C_i)$  value is lower than  $t$ :  $\hat{U}(t) = \{C_i | AC(C_i) < t\}$ . The effectiveness is evaluated by the following indicators.

*Homogeneity*. For each cluster  $C_i$  it represents the portion of patterns belonging to the class assigned to the cluster:

$$homogeneity(C_i) = \frac{|\mathbf{x}_j \in C_i | f(\mathbf{x}_j) = y_{C_i}|}{|C_i|} \quad (9)$$

*Precision*( $t$ ). It represents the portion of clusters in  $\hat{U}(t)$  that are unknown, i.e. that belong to  $U$ :

$$precision(t) = \frac{|\hat{U}(t) \cap U|}{|\hat{U}(t)|} \quad (10)$$

*Recall*( $t$ ). It represents the portion of unknown clusters  $U$  that have been correctly retrieved in  $\hat{U}(t)$ :

$$recall(t) = \frac{|\hat{U}(t) \cap U|}{|U|} \quad (11)$$

*Class recall*( $t$ ) and *class precision*( $t$ ). The same precision and recall indicators described above are computed also at activity class level (rather than cluster) to provide a complementary information (we should in fact consider that each activity class can be mapped to different clusters).

#### 3.2. Results on OAD and ablation study

The aim of the experiments on the OAD dataset is two-fold: on the one hand, an internal evaluation of the proposed approach is performed in an ablation study on the clustering and

<sup>1</sup>OAD dataset.



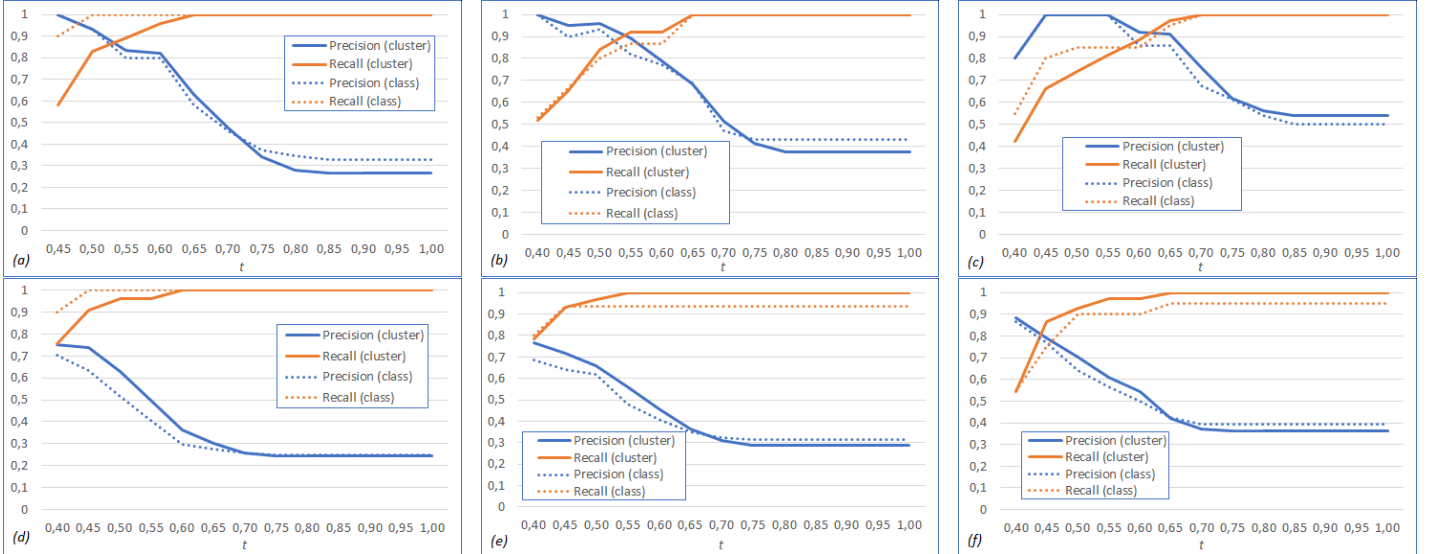


Figure 2: Precision-recall curves (OAD dataset) at cluster and class level, as a function of the threshold  $t$  applied to the *adherence to classes* value  $AC(C_i)$ , used to label clusters  $C_i$  as unknown activities. The graphs refer to a different number of known/unknown activities: 3/2 (a), 3/3 (b), 3/4 (c) 6/2 (d), 6/3 (e) and 6/4 (f).

dimensionality reduction algorithms; on the other, an in depth analysis of the performance obtained in different testing scenarios, i.e. with a variable number of known/unknown activities, is reported. For the experiments on the OAD dataset the neighborhood value used for LOF has been set to 10, considering that for each activity the *test set* contains 30 samples. This leads to the removal of about 10% of the data. For classification a Support Vector Machine (SVM) (Cortes and Vapnik (1995)) with RBF kernel was chosen. To conduct an ablation study, the results of the proposed approach are compared to those obtained by changing:

- *dimensionality reduction*: as an alternative to NCA, we internally evaluated several alternatives (among which the well-known UMAP approach proposed in McInnes et al. (2020)) and the best results have been obtained with Linear Discriminant Analysis (LDA) (Li and Jain (2009)), specifically designed to maximize the class discriminability. Due to the dimensionality constraints w.r.t. the number of classes, the reduced dimensionality has been fixed to 2.
- *clustering algorithm*: the results obtained with Affinity Propagation are compared with those obtained using a density-based solution, i.e. HDBSCAN (Campello et al. (2013)), an extremely robust density-based clustering algorithm, capable of identifying clusters with different density levels.

The results obtained in terms of *homogeneity* on the OAD dataset are summarized in Table 1, which reports the average value of *homogeneity* measured for different testing setups determined by a variable number of known/unknown activities. The table shows the performance for different combinations of dimensionality reduction (NCA 300 dim, NCA 100 dim, LDA) and clustering algorithm (Affinity Propagation vs. HDBSCAN). With regard to dimensionality reduction, NCA (reduced dimensionality 300) generally outperforms the alterna-

tive solutions. As far as clustering is concerned, Affinity Propagation shows a significantly better performance than its competitor. The values obtained are very high, in particular when coupled with NCA 300 where on average around 90% of the data assigned to each cluster belong to a single activity class. A very interesting aspect to consider is that the results are quite constant across the different setups considered, even when the number of unknown activities increases, thus confirming a good robustness of the proposed approach.

Figure 2 shows all the results obtained in terms of precision and recall, as a function of the threshold  $t$  applied to the *adherence to classes* value  $AC(C_i)$ , used to label clusters  $C_i$  as unknown activities. The two curves have, as expected, an inverse trend: higher threshold values allow to increase the recall, i.e. to identify a higher number of unknown activities, but the precision decreases accordingly indicating that some of the clusters considered unknown correspond to a known activity class. In all cases, it's possible to identify a threshold corresponding to an optimal trade-off between the two indicators; the optimal value is generally inversely proportional to the number of known activities: around 0.55 in the test case with 3 known activities, about 0.4 with 6 known activities. This behaviour is reasonable if we consider that the uncertainty of the activity classifier will be generally higher when dealing with a higher number of classes. The optimal threshold identified allows to reach very good precision/recall values, between 85% and 90% with 3 known activities and 2 to 4 unknown, between 75% and 80% with 6 known activities and 2 to 4 unknown. These results are very encouraging since most of the unknown activities can be successfully identified and suggested to the user for a human verification and a semantic meaning assignment. The good results are confirmed when we observe the class-based results, thus allowing to conclude that the approach is able to correctly suggest to the user most of the unknown activities. The choice of the operative threshold to be used can be partially automated according to the results of an internal evaluation, but we believe

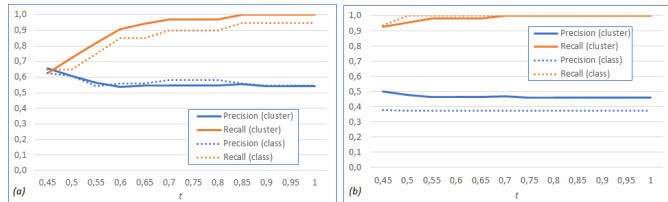


Figure 3: Precision-recall curves (OAD dataset) at cluster and class level, obtained with LDA reduction, as a function of the threshold  $t$  for the adherence to classes  $AC(C_i)$ . The graphs refer to a different number of known/unknown activities: 3/4 (a), 6/4 (b).

Table 1: Average cluster homogeneity on the OAD dataset for different values of known/unknown activities. The results of different dimensionality reduction (NCA vs. LDA) and clustering approaches (Affinity Propagation vs. HDBSCAN) are reported.

Dim. Red.	Known Unknown	3			6		
		2	3	4	2	3	4
NCA (300)	Affinity Prop.	<b>0.92</b>	<b>0.93</b>	<b>0.89</b>	0.87	<b>0.90</b>	<b>0.89</b>
	HDBSCAN	0.72	0.79	0.60	0.77	0.78	0.75
NCA (100)	Affinity Prop.	0.89	0.88	0.88	<b>0.88</b>	0.85	0.78
	HDBSCAN	0.76	0.67	0.70	0.76	0.73	0.79
LDA	Affinity Prop.	0.65	0.65	0.63	0.65	0.60	0.59
	HDBSCAN	0.54	0.54	0.56	0.45	0.48	0.53

that the value has to be tuned by the final user according to the desired precision/recall tradeoff.

For comparative purposes, Figure 3 reports the precision-recall curves obtained using LDA dimensionality instead of NCA for the two testing scenarios of Figure 2(c) and (f). The trend observed for LDA is far less satisfactory; for the first case, the best precision/recall is reached at about 65% while NCA in the same test achieved a value around 90%. For the second testing case (6 known activities, 4 unknown) a good recall value is observed but the precision is constantly low, independently on the threshold selected, thus leading to many wrong suggestions to the user and reducing the usability of the proposed system. The superiority of NCA seems to suggest that it is better suited for data sets of unknown complexity and structure; indeed, unlike LDA, NCA does not make any assumptions about the class distributions. Overall, the results confirm the superiority of the proposed approach and the feasibility of NCA for dimensionality reduction and Affinity Propagation for clustering.

### 3.3. Results on KTH dataset

Further experiments have been conducted on the KTH dataset to evaluate the sensitivity of the proposed approach to one of its main parameters, i.e. the number of neighbors used for outlier removal (see LOF, section 2.2.1). Table 2 shows the results obtained in terms of cluster homogeneity according to the number of known/unknown activities and the number of neighbors considered for outlier removal. The results obtained can be considered quite good. On average, more than 80% of the data assigned to each cluster belong to the same class, even if a slight decrease can be observed as the number of activities increases. Also in this case, the best results were obtained using NCA and a dimensionality equal to 300.

These results overall confirm the effectiveness of the algorithm for outlier detection (LOF) and its robustness with respect

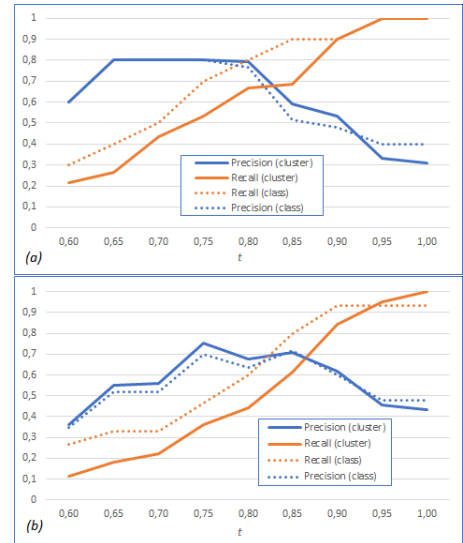


Figure 4: Precision-recall curves (KTH dataset) at cluster and class level, as a function of the threshold  $t$  for the adherence to classes  $AC(C_i)$ . The graphs refer to a different number of known/unknown activities: 3/2 (a), 3/3 (b).

Table 2: Average cluster homogeneity on the KTH dataset for different values of known/unknown activities as a function of the number of neighbors  $k$  used for outlier removal.

# known/unknown activities	# neighbors			
	5	10	15	20
2 - 1	0.85	0.83	0.81	0.84
2 - 2	0.84	0.85	0.86	0.79
3 - 2	0.81	0.82	0.83	0.79
3 - 3	0.75	0.76	0.76	0.77

to its main parameter, i.e. the number of neighbors considered. In fact, Table 2 clearly shows a good stability of the homogeneity value. Figure 4 reports the precision/recall curves for the experiment with 3 known and 2 or 3 unknown activities; the number of neighbors for LOF has been fixed here to 15. The results obtained here are slightly inferior to those observed in the OAD dataset. The algorithm reaches here a trade-off between precision and recall around 70% (80% precision with 70% recall for 3 known and 2 unknown activities). Differently from the OAD dataset, in this case the optimal threshold for the  $AC(C_i)$  value is higher, between 0.80 and 0.85. This particular behaviour together with the lower precision/recall values observed are probably related the features used to encode the activities, which are less effective in this case, thus leading to a higher level of uncertainty of the classifier, even for the set of known activities. At class-level, the precision and recall values obtained are in line or even better than those observed at cluster-level.

## 4. Conclusions

The development of adaptable HAR systems able to exploit continuous learning strategies must necessarily consider an open-set scenario where only a part of the activities performed by the subjects in the environment are known to the system. In this work an approach for identifying unknown activi-

ties has been proposed, based on a combination of metric learning, an advanced clustering technique and ad hoc indicators defined to estimate the membership of each pattern to known classes. The experiments carried out on two public benchmarks confirm the effectiveness of the proposal; precision/recall values around 80% or higher are achieved in the complex OAD dataset, meaning that a high percentage of unknown activities can be successfully suggested to the users with good precision.

In our future work, the main efforts will be devoted to the integration of this approach into a more complete continuous activity learning system where activity detection, unknown activity discovery and automated template updating are effectively integrated and adapted even to large scale scenarios.

## References

- Bai, L., Yeung, C., Efstratiou, C., Chikomo, M., 2019. Motion2vector: Unsupervised learning in human activity recognition using wrist-sensing data, in: Adjunct Proc. of the ACM Int. Joint Conf. on Pervasive and Ubiquitous Computing and of the ACM Int. Symposium on Wearable Computers, Association for Computing Machinery, New York, NY, USA. p. 537–542.
- Beddiar, D.R., Nini, B., Sabokrou, M., Hadid, A., 2020. Vision-based human activity recognition: a survey. *Multimedia Tools and Applications* 79, 30509–30555.
- Breunig, M.M., Kriegel, H.P., Ng, R.T., Sander, J., 2000. Lof: identifying density-based local outliers, in: Proceedings of the 2000 ACM SIGMOD international conference on Management of data, pp. 93–104.
- Campello, R.J., Moulavi, D., Sander, J., 2013. Density-based clustering based on hierarchical density estimates, in: Pacific-Asia conference on knowledge discovery and data mining, Springer. pp. 160–172.
- Chen, L., Lu, J., Song, Z., Zhou, J., 2021. Recurrent semantic preserving generation for action prediction. *IEEE Transactions on Circuits and Systems for Video Technology* 31, 231–245.
- Chen, M.H., Li, B., Bao, Y., AlRegib, G., Kira, Z., 2020. Action segmentation with joint self-supervised temporal domain adaptation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.
- Cortes, C., Vapnik, V., 1995. Support-vector networks, in: *Machine Learning*, pp. 273–297.
- Ehatisham-Ul-Haq, M., Javed, A., Azam, M.A., Malik, H.M.A., Irtaza, A., Lee, I.H., Mahmood, M.T., 2019. Robust human activity recognition using multimodal feature-level fusion. *IEEE Access* 7, 60736–60751.
- Fernando, B., Shirazi, S., Gould, S., 2017. Unsupervised human action detection by action matching, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 1–9.
- Franco, A., Magnani, A., Maio, D., 2017. Joint orientations from skeleton data for human activity recognition, in: *Image Analysis and Processing - ICIAP 2017 - 19th International Conference, Proceedings, Part I*, pp. 152–162.
- Franco, A., Magnani, A., Maio, D., 2020a. A multimodal approach for human activity recognition based on skeleton and rgb data. *Pattern Recognition Letters* 131, 293 – 299.
- Franco, A., Magnani, A., Maio, D., 2020b. Template co-updating in multimodal human activity recognition systems, in: Proceedings of the 35th Annual ACM Symposium on Applied Computing, Association for Computing Machinery, New York, NY, USA. p. 2113–2116.
- Frey, B.J., Dueck, D., 2007. Clustering by passing messages between data points. *science* 315, 972–976.
- Goldberger, J., Hinton, G.E., Roweis, S.T., Salakhutdinov, R.R., 2005. Neighbourhood components analysis, in: *Advances in neural information processing systems*, pp. 513–520.
- Gutoski, M., Lazzaretti, A., Lopes, H., 2020. Deep metric learning for open-set human action recognition in videos. *Neural Computing and Applications* .
- Ihianle, I.K., Nwajana, A.O., Ebeonuwa, S.H., Otuka, R.I., Owa, K., Orisatoki, M.O., 2020. A deep learning approach for human activities recognition from multimodal sensing devices. *IEEE Access* 8, 179028–179038.
- Jaeyong Sung, Ponce, C., Selman, B., Saxena, A., 2012. Unstructured human activity detection from rgb-d images, in: 2012 IEEE International Conference on Robotics and Automation, pp. 842–849.
- Kaya, M., Bilge, H., 2019. Deep metric learning: A survey. *Symmetry* 11.
- Kwon, Y., Kang, K., Bae, C., 2014. Unsupervised learning for human activity recognition using smartphone sensors. *Expert Systems with Applications* 41, 6067 – 6074.
- Li, S.Z., Jain, A. (Eds.), 2009. LDA (Linear Discriminant Analysis). Springer US, Boston, MA. pp. 899–899.
- Li, Y., Shi, D., Ding, B., Liu, D., 2014. Unsupervised feature learning for human activity recognition using smartphone sensors, in: Prasath, R., O’Reilly, P., Kathirvalavakumar, T. (Eds.), *Mining Intelligence and Knowledge Exploration*, Springer International Publishing, Cham. pp. 99–107.
- Lu, Y., Wei, Y., Liu, L., Zhong, J., Sun, L., Liu, Y., 2017. Towards unsupervised physical activity recognition using smartphone accelerometers. *Multimedia Tools Appl.* 76, 10701–10719.
- McInnes, L., Healy, J., Melville, J., 2020. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv:1802.03426*.
- Minh Dang, L., Min, K., Wang, H., Jalil Piran, M., Hee Lee, C., Moon, H., 2020. Sensor-based and vision-based human activity recognition: A comprehensive survey. *Pattern Recognition* 108, 107561.
- Moschetti, A., Fiorini, L., Esposito, D., Dario, P., Cavallo, F., 2017. Daily activity recognition with inertial ring and bracelet: An unsupervised approach, in: 2017 IEEE International Conference on Robotics and Automation (ICRA), pp. 3250–3255.
- Ong, W.H., Koseki, T., Palafox, L., 2013. Unsupervised human activity detection with skeleton data from rgb-d sensor, in: *Int. Conference on Computational Intelligence, Communication Systems and Networks*, pp. 30–35.
- Perš, J., Sulčić, V., Kristan, M., Perše, M., Polanec, K., Kovačič, S., 2010. Histograms of optical flow for efficient representation of body motion. *Pattern Recognition Letters* 31, 1369–1376.
- Riboni, D., Szttyler, T., Civitarese, G., Stuckenschmidt, H., 2016. Unsupervised recognition of interleaved activities of daily living through ontological and probabilistic reasoning, in: Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing, Association for Computing Machinery, New York, NY, USA. p. 1–12.
- Schuldts, C., Laptev, I., Caputo, B., 2004. Recognizing human actions: a local svm approach, in: Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004., IEEE. pp. 32–36.
- Song, H.O., Xiang, Y., Jegelka, S., Savarese, S., 2016. Deep metric learning via lifted structured feature embedding, in: *Computer Vision and Pattern Recognition (CVPR)*.
- Su, K., Liu, X., Shlizerman, E., 2020. Predict cluster: Unsupervised skeleton based action recognition, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society, Los Alamitos, CA, USA. pp. 9628–9637.
- Tang, Y., Lu, J., Wang, Z., Yang, M., Zhou, J., 2019. Learning semantics-preserving attention and contextual interaction for group activity recognition. *IEEE Transactions on Image Processing* 28, 4997–5012.
- Tang, Y., Lu, J., Zhou, J., 2020a. Comprehensive instructional video analysis: The coin dataset and performance evaluation. *IEEE Transactions on Pattern Analysis Machine Intelligence* , 1–1doi:10.1109/TPAMI.2020.2980824.
- Tang, Y., Wei, Y., Yu, X., Lu, J., Zhou, J., 2020b. Graph interaction networks for relation transfer in human activity videos. *IEEE Transactions on Circuits and Systems for Video Technology* 30, 2872–2886.
- Trabelsi, D., Mohammed, S., Chamroukhi, F., Oukhellou, L., Amirat, Y., 2013. An unsupervised approach for automatic activity recognition based on hidden markov model regression. *IEEE Transactions on Automation Science and Engineering* 10, 829–835.
- Wang, H., Schmid, C., 2013. Action recognition with improved trajectories, in: Proc. of the IEEE international conference on computer vision, pp. 3551–3558.
- Wang, H., Ullah, M.M., Klaser, A., Laptev, I., Schmid, C., 2009. Evaluation of local spatio-temporal features for action recognition.
- Wang, P., Li, W., Ogunbona, P., Wan, J., Escalera, S., 2018. Rgb-d-based human motion recognition with deep learning: A survey. *Computer Vision and Image Understanding* 171, 118 – 139.
- Xu, R., Wunsch, D., 2005. Survey of clustering algorithms. *IEEE Transactions on neural networks* 16, 645–678.
- Yu, J., Kim, D.Y., Yoon, Y., Jeon, M., 2020. Action matching network: open-set action recognition using spatio-temporal representation matching. *The Visual Computer* 36.
- Zheng, W., Lu, J., Zhou, J., 2020. Hardness-aware deep metric learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence* , 1–1.