Soft Comput (2015) 19:2435–2445
DOI 10.1007/s00500-014-1499-y

FOCUS

Unsupervised template discovery in activity recognition using the Gamma Growing Neural Gas algorithm

Héctor F. Satizábal · Andres Perez-Uribe

Published online: 2 November 2014

Abstract Activity recognition is gaining a lot of interest given its direct use in applications like ambient assisted living and has been empowered by the increasing ubiquity of sensors (e.g., clothes, smartphones, watches). The machine learning approach to activity recognition consists on finding the signatures characterizing the activities to be recognized, with the hope of identifying them (pattern matching) within the stream of sensor data. The finding of those signatures can be very complex, thus many approaches deal with the streams of sensor data by segmenting them into sections or "time-windows", before processing them by a feature extraction procedure. The problem then concerns the association of features to class labels. In this paper, we propose the use of the Gamma Growing Neural Gas algorithm to unsupervisely discover templates in a recording containing gestures performed by a person in a home environment. The system is able to do vector quantization from the time-series of data coming from one accelerometer, and finds salient patterns (e.g., templates) in the signal. These templates integrate information not only from single time-windows but do consider the recent history of the incoming signal (e.g., multiple time-windows). Those templates are then associated to activity classes by supervised learning. Our experiments show that the resulting performance is better than previous benchmarks of the same database.

Communicated by I. R. Ruiz.

H. F. Satizábal (✉) · A. Perez-Uribe
IICT, HEIG-VD, University of Applied Sciences Western Switzerland (HES-SO), Delémont, Switzerland
e-mail: hector-fabio.satizabal-mejia@heig-vd.ch

A. Perez-Uribe
e-mail: andres.perez-uribe@heig-vd.ch

Keywords Activity recognition · Growing Neural Gas · Time series · Template matching

1 Introduction

Activity and context recognition has become a prolific field in the last years due to the increasing availability of portable and embedded sensors (Lara et al. 2012; Sagha et al. 2011; Stiefmeier et al. 2008). Smartphones, watches, glasses and even clothes can be endowed with sensing capabilities allowing the identification of the activities being performed and the inference of the context in which they are realized. There are many potential applications for activity recognition: home applications like ambient assisted living (Hondori et al. 2012), industrial applications like human-computer interaction and video surveillance (Stiefmeier et al. 2008), and activity life logging with its applications in health, sports and well-being (Rawassizadeh et al. 2013). Our application domain being ambient assisted living, we have oriented our literature review towards the domains of health and elderly assistance. For instance, Amft and Tröster (2008) used a variety of on-body sensors to perform dietary monitoring and therefore help patients with their nutrition. They used multiple modalities (accelerometers, microphones and electromyograms) to detect various activities like chewing, swallowing, drinking, cutting, eating soup. Hondori et al. (2012) presented a system that helps monitor various dining activities of post-stroke patients using a Kinect camera and accelerometers. Kepski and Kwolek (2012) built a system that uses the Kinect and a single accelerometer to perform fall detection. Our overall motivation is the use of activity recognition to help people with memory problems. We envision, for instance, the development of a log[1] of high-level daily activities (e.g., eating, cleaning, drinking, reading) where a person can browse what he/she has done during the day or week. Such a system can help people by acting as a cognitive compensation system, and at the same time, given that memory loss can be frustrating, it can provide a positive feedback letting them remember their recent activities (Browne et al. 2011).

The traditional approach to build an activity recognition application implies a lot of training effort. The processing chain (Bulling et al. 2014) of the sensor data starts with the sensor data acquisition: a stream of sensor samples is obtained; the sensor data stream is then preprocessed: typical transformations are calibration, de-noising, or sensor level data fusion. Then, the data stream is segmented into sections. A common type of segmentation technique is the sliding window. To characterise these raw data and reduce their dimensionality, features are computed on the identified segments. A classifier, trained at design-time, maps the feature vector into a pre-defined set of output classes (e.g., activities).

We can distinguish two main approaches to the characterisation of the input signals. The first approach consists on using information from a single time-window, while the second approach consists on using information extracted from subsequent multiple time-windows, in the form of time series. In this work, we start by presenting an overview of these two approaches. We present the advantages and disadvantages, as well as a literature review of recent works having used both methods. The central contribution of this paper is the introduction of the application of a time-series vector quantization algorithm, called the Gamma Growing Neural Gas algorithm (γ-GNG), within the standard activity recognition chain. The advantage of using this technique is that it allows us to deal with multiple subsequent time-windows of the stream of sensor data, thus conveying more information to the machine learning classifiers.

The article is organized as follows. Section 2 describes recent related works in the field, emphasizing on whether they use information from single time-windows or from multiple time-windows to infer the activities they want to recognize. Section 3 presents a detailed explanation of the γ-GNG algorithm approach for template discovery in time series. Section 3.1 describes the algorithm itself and Sect. 3.2 the experimental setup we used to analyse the behaviour and the performance of the algorithm. The results and discussion of the tests are given in Sect. 3.3. Finally, Sect. 4 presents our conclusions.

2 Related work

The activity recognition approaches related to our work can be classified in two groups: (i) techniques that use information from single time-windows of the sensor data, and (ii) techniques that use sequences of data containing the recent history of the input signals. Table 1 presents a list of recent works in the field of activity recognition, classified into the aforementioned categories. In the following, we will briefly describe the advantages and disadvantages of both approaches.

The first approach considers only sensor data from a given period of time and does not take into account the information of what has just happened. This approach is computationally cheap and has been successfully exploited on the recognition of simple activities (Delachaux et al. 2013). However, a single time-window cannot fully characterise the activity that is being performed and in most cases, other sources of information (sensors) are needed to identify a particular activity from a plurality of different activities. Moreover, the size of the time-window from which features are computed plays an important role in the classification results. For short time-windows, this approach is equivalent to infer the activities or gestures performed by a person from his/her posture. For long time-windows, this approach produces good results only if the targeted activity is uniform within the window, and the features computed from the window capture the dynamics of the activity within this period of time (e.g., frequency domain features).

The second approach considers sequences of multiple time-windows, which is more complex than using information from a single window. For instance, the optimal duration of the time-windows from where features are computed, which are the most informative features, and the optimal length of the sequences are open questions. Moreover, the techniques for comparing sequences are computationally expensive. However, there are some advantages of taking into account the recent history of the signal as input to the classifiers. Given that more information is used to characterise the activities, often less sensors and features[2] are needed to obtain the same results (Satizábal et al. 2013).

One straightforward and very widely used approach for processing time series is template matching. The goal of template matching is to detect some salient patterns in the time series, and to use these detections to infer the occurrence of a particular activity. A set of templates representing these activities is therefore needed. These templates can be created in a synthetic manner or, in a more adaptable way, they can be extracted from the signal itself using clustering techniques. For instance, Satizábal et al. (2013) explored the use

[1] A collection of snapshots of the person or images of an avatar performing the activity.

[2] Since a longer portion of the signal is available, no features characterizing the dynamics of the signal within long time-windows are needed.

Table 1 Recent works on activity recognition classified according to whether they use information from a single time-window or sequences of data containing the recent history of the signals

References	Class	Application	Sensors
Hartmann and Link (2010)	ii	Industry	U
Lukowicz et al. (2010)	–	Home	U
Roggen et al. (2010)	i	Home	U + O
Xue and Jin (2010)	i	–	U
Aggarwal and Ryoo (2011)	ii	Survey	A
van Kasteren et al. (2011)	ii	Home	O
Kasteren et al. (2011)	ii	Home	O
Sagha et al. (2011)	i	Home	U
Baños et al. (2012)	i	Fitness	U
Chen et al. (2012)	–	Survey	–
Hondori et al. (2012)	i	Home	U + A
Kepski and Kwolek (2012)	i	Home	U + A
Lara et al. (2012)	i	Home	U
Nguyen-Dinh et al. (2012)	ii	Home	U
Baños et al. (2013)	i	Fitness	U
Chavarriaga et al. (2013)	i	–	U
Delachaux et al. (2013)	i	Home	U + A
Leppanen and Eronen (2013)	i	–	U
Ni et al. (2013)	–	Home	A
Rawassizadeh et al. (2013)	i	Life-log	U
Rebetez et al. (2013)	i	Home	U
Satizábal et al. (2013)	ii	Home	U
Shotton et al. (2013)	i	–	A

U wore by the user, *A* in the ambient and *O* embedded in objects

of a semi-supervised approach for finding gesture "fingerprints" that are further used as templates to perform gesture spotting. They used a two-step approach in which (i) the incoming sequences are grouped in an unsupervised manner to find good candidates to gesture templates, and (ii) the candidates are evaluated in a supervised manner by comparing them with the ground truth (i.e., labels in the database). In the first step, they used clustering techniques to find a set of template candidates in an unsupervised manner. They adapted the clustering algorithm (i.e., k-medoids) by embedding diverse distance measures to compare the sequences. They evaluated the performance of the resulting classifiers using the Euclidean distance, the dynamic time warping (DTW) distance and the longest common subsequence (LCSS) distance, and concluded that the best results were obtained using the DTW distance, and running the algorithm several times with different sequence lengths. The DTW distance has shown to be robust to small local variations in the speed of the sequences, thus allowing the comparison of time series that are similar but locally out of phase, at the expense of being computationally expensive.

The approach we propose in this contribution belongs to this latter category but, contrary to more conventional methodologies, it allows to compare longer signal segments using a shorter buffer size. Moreover, for the test we performed, it yields better results even using a less computationally expensive distance measure (i.e., weighted Euclidean distance instead of DTW).

3 Unsupervised template discovery

In this section we present a vector quantization approach for finding repetitive patterns within a time series (templates). The proposed approach has a memory with variable resolution that allows the buffer to store longer segments of the signal at the expense of the accuracy of the comparisons.

The approach we describe here is based on the use of a well-known vector quantization neural network in which a memory has been added. This memory allows the network to keep information about the history of the signal (context), and to find prototypes that take into account the temporal context of it.

3.1 The Gamma Growing Neural Gas algorithm

The Gamma Growing Neural Gas (γ-GNG) algorithm was proposed by Estévez and Hernández (2011) as a new approach for processing data sequences that are temporally or

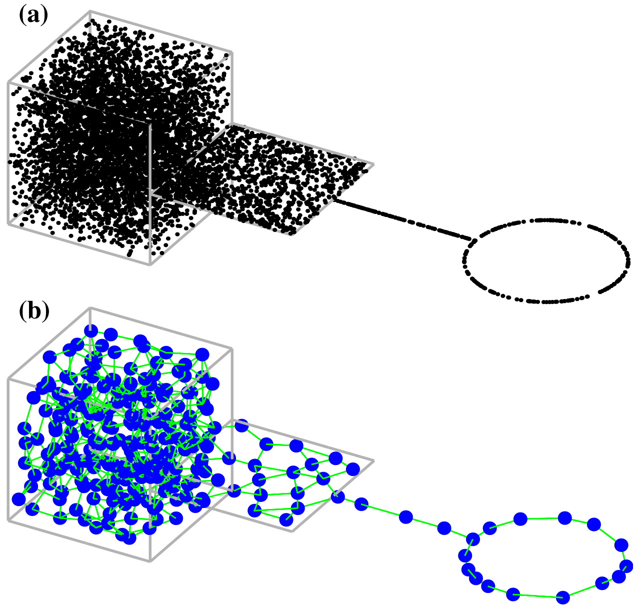


Fig. 1 Example of the distribution of prototypes generated by the Growing Neural Gas algorithm. **a** The dataset, **b** the resulting distribution of prototypes

spatially connected e.g., words, DNA sequences, time series. It merges the standard GNG algorithm (Fritzke 1995) with a context descriptor based on a short term memory structure called Gamma filter (de Vries and Principe 1992; Principe et al. 1993). This subsection explains some generalities of the two main components of the γ-GNG algorithm i.e., The GNG algorithm and the Gamma filter, and how these components build the algorithm.

3.1.1 The Growing Neural Gas algorithm

The GNG (Fritzke 1995) is an incremental neural network model which performs vector quantization and topology learning. The algorithm starts with two nodes and incrementally builds a network of prototypes by adding units using a competitive Hebbian learning strategy. New prototypes are added in regions where the error counter of each node is high. Old edges between nodes are broken if their age is higher than a threshold, and the resulting isolated nodes are deleted. The resulting structure is a graph of prototypes that reproduces the topology of the input dataset by keeping the distribution and the dimensionality of the data. Figure 1 shows an example of the resulting prototypes after applying the GNG algorithm to a synthetic dataset.

As it can be seen in Fig. 1, the distribution of prototypes inserted by the GNG algorithm keeps the distribution and dimensionality of the input data. Hence, each one of these prototypes represents one small region in the input space, and adding more prototypes makes the quantization more precise.

3.1.2 The Gamma filter

The γ-GNG model is a merge of the GNG algorithm (Fritzke 1995) and a memory based on the Gamma filter (de Vries and Principe 1992; Principe et al. 1993). The whole γ-GNG network, as well as each one of its prototypes, keeps a record of the recent history of the incoming signal. This record is called the “context” of the signal.

$$\begin{aligned} y(n) &= \sum_{k=0}^{K} \omega_k c_k(n) \\ c_k(n) &= \beta c_k(n-1) + (1-\beta)\, c_{k-1}(n-1) \end{aligned} \tag{1}$$

The Gamma filter (de Vries and Principe 1992) is defined in the time domain as it is shown in Eq. 1. Where $c_0(n) \equiv x(n)$ is the input signal, $y(n)$ is the filter output, $w_0, ..., w_K$, β are the filter parameters and K is the order of the filter. The $\beta \in (0, 1)$ parameter provides a mechanism to decouple depth (D) and resolution (R) from filter order. Depth measures how far into the past the memory stores information, thus a low memory depth can hold only more recent information. Resolution indicates the degree to which information concerning the individual elements of the input sequence is preserved. The mean memory depth for a Gamma filter of order K is shown in Eq. 2, and its resolution is shown in Eq. 3.

$$D = \frac{K}{(1-\beta)} \tag{2}$$

$$R = 1 - \beta \tag{3}$$

Each node i of the γ-GNG model has a vector prototype $w^i \in \Re^d$ computed using the GNG algorithm (vector quantization). Additionally, each node i in the network has a set of contexts $C = \{c_1^i, c_2^i, ..., c_K^i\}$, $c_k^i \in \Re^d$, $k = 1, ..., K$.

Moreover, given a sequence entry $x(n)$, the best matching unit, I_n, is the neuron that minimizes the distance criterion shown in Eq. 4. The parameters α_ω and α_k, $k \in \{1, 2, ..., K\}$ control the relevance of the different elements i.e., α_ω is the weight of the position and α_k is the weight of the context.

$$d_i(n) = \alpha_\omega \|x(n) - \omega^i\|^2 + \sum_{k=0}^{K} \alpha_k \|c_k(n) - c_k^i\|^2 \tag{4}$$

To compute the recursive distance 4, a context descriptor is required in the different filtering stages. The K context descriptors of the network are defined as shown in Eq. 5. Where $c_0^{I_{n-1}} \equiv \omega^{I_{n-1}}$ and at $n = 0$ the initial conditions $c_k^{I_0}, \forall k = 1, ..., K$ are set randomly, where K is the filter order.

$$c_k(n) = \beta c_k^{I_{n-1}} + (1-\beta) c_{k-1}^{I_{n-1}} \forall k = 1, ..., K \tag{5}$$

Because the context construction is recursive, it is recommended that $\alpha_\omega > \alpha_1 > \alpha_2 > \cdots > \alpha_K > 0$, otherwise errors in the early filter stages may propagate through higher-order contexts. An example of how to compute these α_i parameters is shown in Eq. 6.

$$\alpha_i = \frac{K+1-i}{\sum_{k=0}^{K}(k+1)}, \quad i = 0, ..., K \tag{6}$$

Notice that Eqs. 4 and 6 represent a weighted Euclidean distance where the older context descriptors have less weight than the more recent ones. This distance measure works well for short length vectors, otherwise it could be worth to use a more specialized distance measure like dynamic time warping (Satizábal et al. 2013).

Summarizing, these are the modifications in the γ-GNG model compared to the original GNG algorithm:

- Besides their positions, each unit in the network has K vectors of context descriptors c_k, where K is the order of the filter.
- The c_k context descriptors of the network are computed using Eq. 5.
- Best matching units are found using Eq. 4.
- The position and context of winner and neighbouring units are modified using the constants ϵ_w and ϵ_n of the original algorithm.

The detailed description of the algorithm is given in (Estévez and Hernández 2011). For the sake of exemplification, we run a test of the γ-GNG algorithm over a dataset captured from a table tennis training session. The dataset contains the acceleration and angular speed captured with an inertial measurement unit[3] (IMU) located at the right wrist of one of the players. Figure 2a shows the signature of two strokes: forehand and backhand.

The objective of the test is to show how the γ-GNG algorithm is able to detect these two strokes from the whole set of strokes executed by the player during the training session.[4] After running the algorithm we computed the activation of the nodes within the window of time where the strokes were executed, i.e., [−0.6, 0.6] seconds. The activation of a given γ-GNG node or prototype occurs when the history of X, Y and Z angular speeds captured by the sensor is very similar to the multi-dimensional time series of angular speeds represented by that node. Figure 2b shows the frequency of activation of the nodes in the network during the strokes, computed by counting the activations of each unit in the network around the moment of impact with the ball. The higher the frequency of activation of a node, the darker the pixel displayed for the corresponding node index. As it can be seen from Fig. 2b, the γ-GNG nodes 46, 37 and 22 are the ones that are more frequently activated during the execution of a forehand gesture, and that the γ-GNG nodes that are more frequently activated during the execution of a backhand gesture are different (i.e., they are the nodes 33, 40, and 45). This shows that it is possible to find nodes that are frequently activated during the execution of a given gesture and not during the execution of other gestures (or activities).

A similar example is shown in Fig. 3. In this case we run the γ-GNG algorithm over a dataset gathered from a person performing typical gestures of daily living activities. The dataset included some features (i.e., mean and angle) computed from the acceleration signal of the wrist of a person.

Figure 3 shows two chunks of the signal, and the corresponding sequence of activation of nodes. Notice that nodes 26 and 47 get activated near the middle and near the end of the gesture, respectively.

3.2 Experimental setup

This section describes the steps we carried out to test how the γ-GNG behaves in the task of finding gesture/activity templates. The tests are not complete in the sense that the diversity of activities performed by a person is huge, and it is very difficult to test all of them. However, we observed the behaviour of the algorithm while varying its parameters to better asses the implications they may have in our application. The whole description of the experiment is detailed in Table 2.

3.2.1 Dataset

The OPPORTUNITY dataset for human activity recognition from wearable, object, and ambient sensors is a dataset devised to benchmark human activity recognition algorithms (classification, automatic data segmentation, sensor fusion, feature extraction, etc.). The original OPPORTUNITY dataset was acquired from 12 subjects while they were performing morning activities and includes 72 sensors of 10 modalities in 15 wireless and wired networked sensor systems in the environment, objects and the body (Roggen et al. 2010). However, only a subset of this dataset containing the recordings of 4 people is available from the UCI machine learning repository (Bache and Lichman 2013). For each one of the 4 subjects there are five daily activity sessions (ADL) and one drill session which has about 20 repetitions of some pre-defined actions. We decided to use only one sensor located on the right lower arm (RLA), and we computed two features from small windows of time:

[3] The IMU includes a three-axial accelerometer and a three-axial gyroscope sampled at 102.4 Hz.

[4] The dataset contains the execution of other strokes e.g., forehand and backhand topspin, forehand and backhand block.

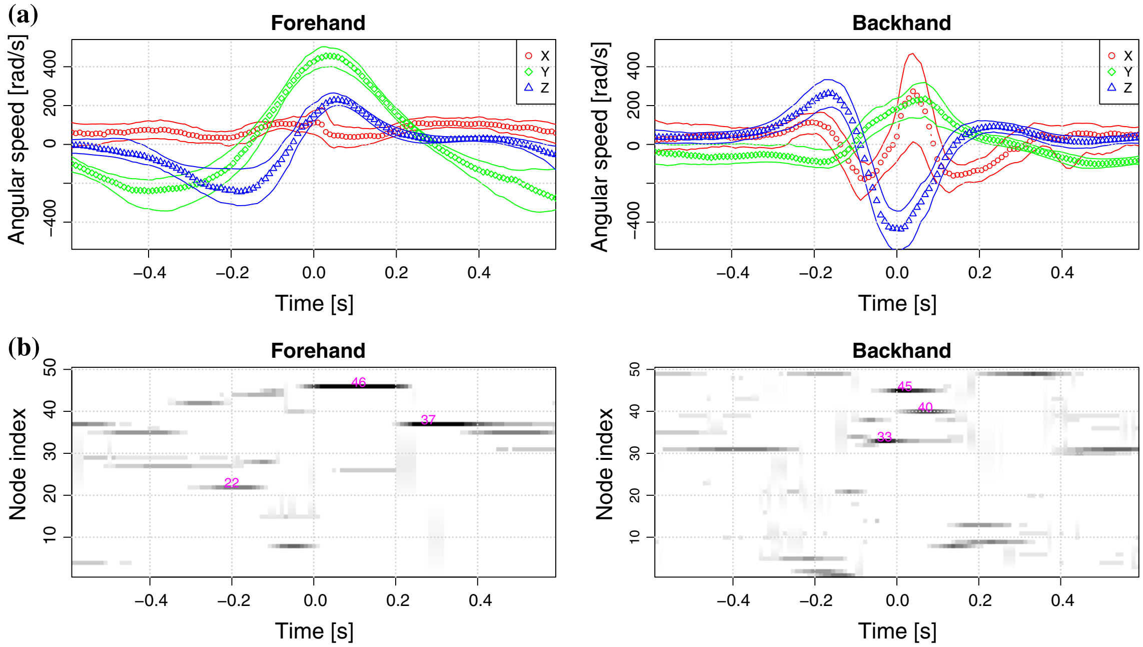


Fig. 2 **a** Signature of two different table tennis strokes. The IMU was located in the right wrist of a right-handed table tennis player. The different repetitions of the strokes where synchronized around the moment of impact with the ball (time = 0). *Dotted lines* represent the median and the *continuous lines* represent the first and third quartile. **b** Frequency of activation of the units in the γ-GNG network. The frequency was computed by counting the activations of each unit in the network around the moment of impact with the ball

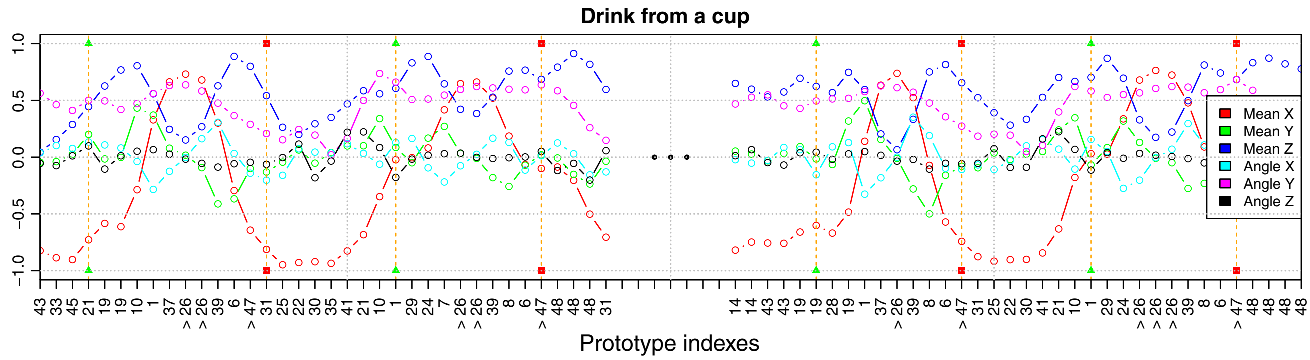


Fig. 3 The n-dimensional time series is represented by a sequence of prototypes. The *horizontal axis* shows the index of the prototype in the network that best matches the signal at each time. *Orange vertical lines* indicate the start (*green triangle*) and stop (*red square*) of four occurrences of the gesture: "Drink from a cup"

(i) The average of the values within the window.
(ii) The angle of the best line segment fitting the data within the window.

Additionally, we tested different window lengths overlapped 50 %.

3.2.2 Model parameters

We performed tests with different model parameters:

(i) Network size: $from = 50$, $to = 500$, $step = 50$.
(ii) Filter order: 1, 5, 10.
(iii) Resolution: 0.9, 0.5 (β: 0.1, 0.5).

The remaining parameters are shown in Table 3.[5] They were set according to previous tests.

[5] See (Estévez and Hernández 2011) for the description of each parameter.

Table 2 Steps in the experimental setup we used for testing the γ-GNG algorithm

Step 1:	Run the γ-GNG algorithm on the whole dataset and compute the weights and contexts of all the prototypes in the network
Step 2:	Compute the sequence of activation of the prototypes in the network by feeding the whole dataset and computing the BMU after each input sample
Step 3:	Split the training dataset and the sequence of activations in 5 parts to perform fivefold cross-validation
Step 4:	For each chunk i of data:
	(a) Create a training and a validation dataset. The training dataset is composed of all the chunks but chunk i; the validation dataset is hence chunk i
	(b) Using the training data, compute the matrix of probability $P(g\|a)$, where $g \in G$, is a particular label (gesture) in the dataset, G is the set of labels, $a \in N$ is the activation of a particular prototype and N is the codebook of the network
	(c) Using the matrix of probability obtained in the previous step, compute the sequence of most probable labels for the sequence of prototypes activations in training and in the validation dataset
	(d) Using the activations obtained in the previous step, compute the F-1 score for the training and the validation dataset

Table 3 Parameters of the γ- GNG algorithm used in the experiments with the OPPORTUNITY dataset

ϵ Best:	0.05	ϵ Neighbors:	0.005
λ:	1,000	a_{max}:	100
$\tilde{\alpha}$:	0.5	$\tilde{\beta}$:	0.01
α:	as in Eq. 6		

3.2.3 Supervised association of gestures to prototypes

The vector prototypes found by the γ-GNG algorithm are located along the distribution of the input data. Given that these prototypes have also been placed according to the local history (context) of the signal, they can be considered as the patterns or building blocks of it. However, for these prototypes to be meaningful, they need to be linked to a particular gesture or "ground truth". We used a probabilistic approach for assigning ground truth labels to each one of the prototypes in the network during an initial training phase:

- We trained the neural network to find the set of prototypes for the time series.
- Once the network is trained, we found the sequence of prototypes that are activated[6] when the dataset is fed into the network. Figure 3 illustrates this transformation.
- Using a dataset with gesture/activity labels at each time, we computed the probability $P(g|a)$ of having a particular gesture/activity given the activation of a prototype.
- The matrix $P(g|a)$ allowed us to infer which gesture the user performed given the activation of each one of the prototypes.

3.2.4 Performance assessment

We evaluated the performance of the whole methodology i.e., unsupervised discovery of templates and supervised association to gestures, by computing the weighted average (over all classes) $F1$-score (Rijsbergen 1979) in the task of classifying the activities in the Drill sessions (without the null class.)[7] The $F1$-score shown in Eq. 7 estimates how accurately our approach classifies a particular activity as such. Where, precision $= \frac{tp}{tp+fp}$, recall $= \frac{tp}{tp+fn}$, tp are the true positives, fp are the true negatives, and fn are the false negatives.

$$f(1) = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{7}$$

3.3 Results and discussion

We tested the γ-GNG algorithm with different window lengths, different features, different filter orders, and different depths/resolutions. Table 4 shows these results.

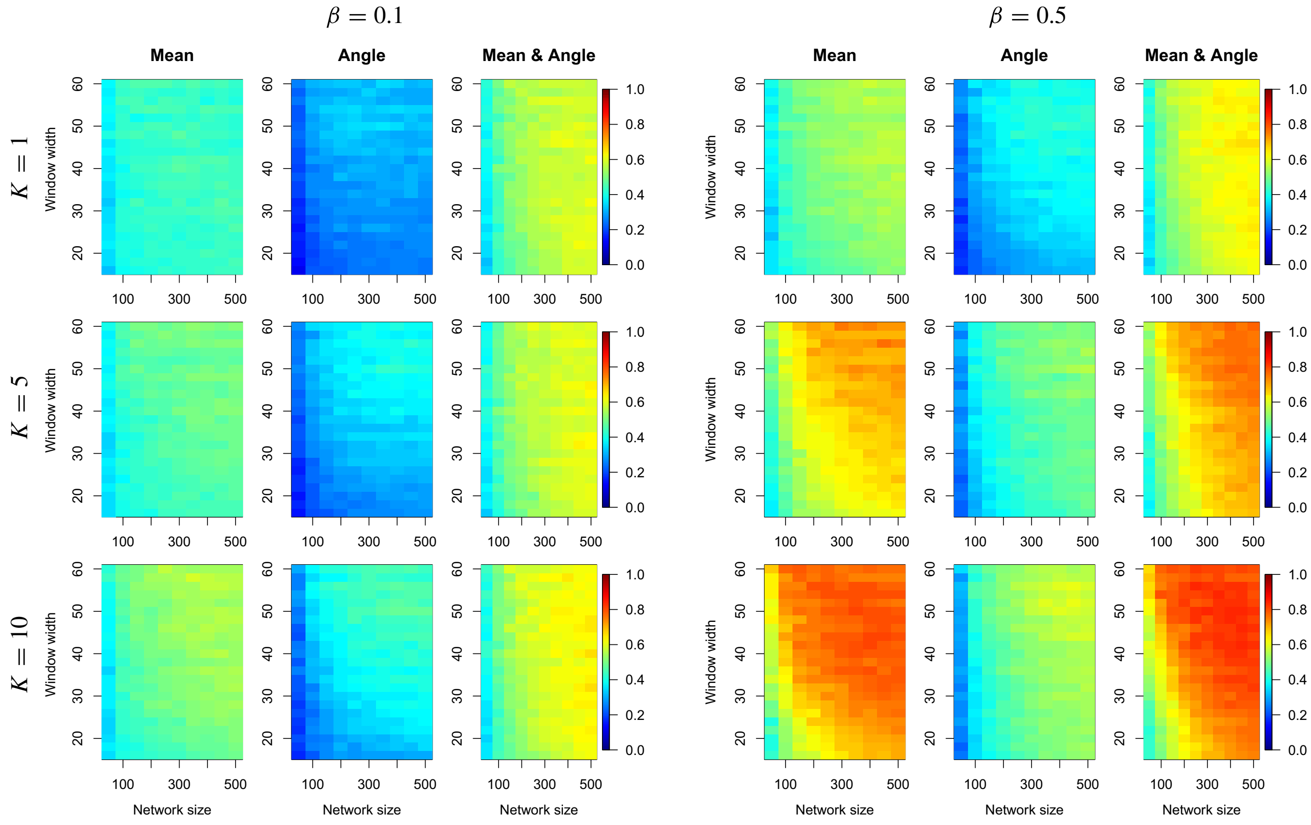
As it can be seen in Table 4, using both features i.e., mean and angle, yields the best results in terms of $F1$-score. However, using both features only produces a minor improvement in the performance which was already high using the average as feature.

Filter order has also a big influence on performance. This result shows that, as expected, the size of the context (and thus the size of the signatures) has a major influence on the classification performance. However, since filter depth can be modulated by parameter β, the length of the context (filter order) does not have to be equal to the length of the gestures. In our experiments parameter β was set to increase depth 1.111 and 2 times (i.e., $\beta = 0.1$ and $\beta = 0.5$) at the expense of decreasing the resolution of the comparisons. One can see in Table 4 that increasing depth ($\beta = 0.5$) produces a better detection of the gestures and thus, higher values of $F1$-score.

[6] We computed the best matching unit (BMU) for each point in the dataset.

[7] The performance of the detection of the null class was not evaluated since this class is much more frequent in the dataset. The performance of the system is overestimated if the null class is considered in the evaluation of the weighted $F1$-score.

Table 4 Cross-validation results for the drill session of subject S1



Different filter orders (i.e., 1, 5, 10) and different beta values (i.e., 0.1 and 0.5) were tested. The colours represent the $F1$-score (without null class) from 0 (*blue*) to 1 (*red*). The horizontal axis indicates the number of prototypes in the network, and the vertical axis indicates the size of the window used for computing the features

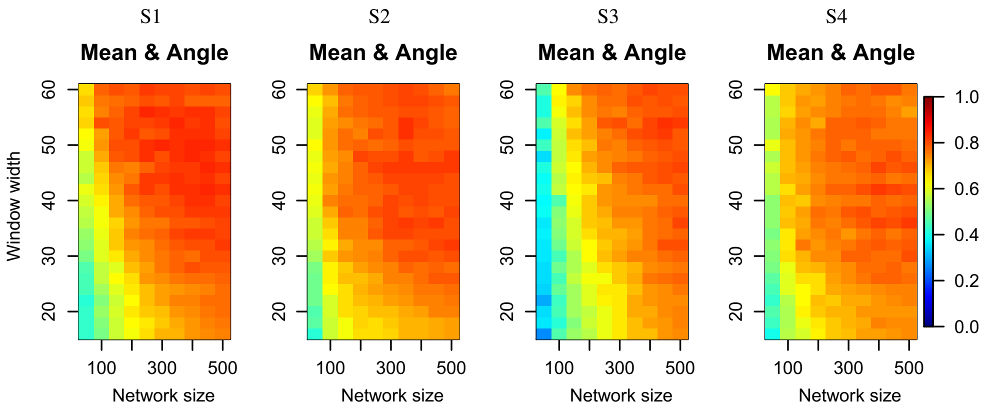
Moreover, the size of the network (the amount of prototypes, and thus the amount of patterns) influences the detection of gestures too. A bigger network (more complex) has more accurate patterns, which produces better detections. Thus, a higher complexity can yield better results, but can be infeasible if the approach has to be implemented in a system with low computational resources (e.g., mobile platforms). Another interesting observation is that the number of units that should have the network for reaching a given $F1$-score changes with the size of the window from where features were computed. If features are computed from larger windows, less units are needed in the network. This result can be explained by the fact that shorter windows generate longer sequences (in terms of number of windows needed to cover them) and thus, given that filter depth is fixed (i.e., filter order and β), more prototypes are needed to cover the full variability of the gestures.

After exploring different parameters for the filter, we concluded that a filter depth of 20 (i.e., $K = 10$, $\beta = 0.5$) gave the best results. Table 5 shows the $F1$-score after cross-validation tests on the 4 subjects in the OPPORTUNITY dataset.

As it can be seen from Table 5, the set of parameters (window width and network size) that yields good results is large. Moreover, the images suggest that comparable performances can be reached adding more units to the γ-GNG network or computing the features using larger time-windows.

For the sake of comparison, we show in Table 6 the performance obtained using other approaches to gesture recognition over the same dataset. Sagha et al. (2011) performed a benchmarking of the dataset using different classifiers processing data from single windows: linear discriminant analysis (LDA), quadratic discriminant analysis (QDA), k Nearest Neighbours (k-NN) and nearest centroid classifier (NCC). Satizábal et al. (2013) proposed an approach that compares sequences of windows based on the k-medoids algorithm by embedding different distance measures: Euclidean distance and the dynamic time warping (DTW) distance.

Table 5 Cross-validation results for the 4 subjects in the OPPORTUNITY dataset



We fixed the filter depth to 20 ($K = 10$, $\beta = 0.5$). The colours represent the $F1$-score (Without null class) from 0 (*blue*) to 1 (*red*). The horizontal axis indicates the number of prototypes in the network, and the vertical axis indicates the size of the window used for computing the features

Table 6 Validation performance (weighted $F1$-score without null class) after applying different methods for recognizing gestures in the OPPORTUNITY dataset

Classifier	S1	S2	S3	S4
LDA†	0.34	0.26	0.33	0.19
QDA†	0.32	0.25	0.39	0.19
1-NN†	0.53	0.47	0.62	0.47
3-NN†	0.52	0.49	0.62	0.48
NCC†	0.30	0.21	0.29	0.15
k-medoids + Euclidean‡	0.46	0.23	0.35	0.58
k-medoids + DTW‡	0.59	0.51	0.5	0.51
γ-GNG (100)	0.64	0.74	0.46	0.69
γ-GNG (200)	0.77	0.76	0.67	0.73
γ-GNG (300)	0.79	0.80	0.73	0.75
γ-GNG (400)	0.82	0.81	0.73	0.77
γ-GNG (500)	0.80	0.80	0.77	0.78

Results marked with † where reported by Sagha et al. (2011). Results marked with ‡ were computed using the approach described by Satizábal et al. (2013). Results of the γ-GNG algorithm were computed using as features the mean and angle of windows of 40 samples overlapped 50 %, a filter depth of 20 and the parameters shown in Table 3

4 Conclusions

The general approach to activity recognition roughly consists on identifying salient sensor data patterns that serve as a "fingerprint" of each activity we want to recognize. One normally looks for those patterns in the input signal after segmenting it into sections or "time-windows". Those patterns are generally multidimensional: i.e., they correspond to particular values of acceleration (and other measures) on every axis, or particular feature values computed on the raw sensor data. A computationally "cheap" approach has been to identify those activity "fingerprints" within single "time-windows". However, when dealing with complex activities, a single time-window cannot fully characterise the activity, and more information is required. One can normally use several sensors to attempt to enrich the fingerprints of the activities, or consider multiple subsequent time-windows. This paper starts by presenting an overview of recent approaches that are based, whether on single time-windows or on sequences of time-windows. It then presents the central contribution, which is the application of a vector quantization algorithm capable of exploiting multiple time-windows of sensor data in activity recognition. This approach integrates the unsupervised discovery of gesture templates (e.g., the activity "fingerprints") into the standard activity recognition chain. The proposed methodology employs an unsupervised neural network called the γ-GNG network which performs vector quantization in feature and "context" spaces. This neural network model keeps a track of the recent history of the signal ("context") by using a filter structure called the Gamma filter. This filter is characterized by (a) an order, which defines its complexity and the amount of time it is able to see into the past of the signal, and (b) a parameter (i.e., β), which can be used to modulate the depth of the memory, at the expense of resolution. We performed a study of the parameters of the γ-GNG algorithm, by analyzing the effect of those parameters, i.e., the length of the windows, the network size and the depth of the filter (by varying its order and resolution) on the performance of the model in the task of activity recognition. These tests were performed using the OPPORTUNITY dataset (Roggen et al. 2010), which is a well-known benchmark in the domain. The results of our tests allowed us to corroborate some straightforward relationships like the effects of the filter order, and the size of network, on the behaviour of the algorithm. Moreover, they allowed us to analyse the less evident relationship between network size and window length, and the effects of the parameter β on the global performance of the system. Last but not least, they allowed us to show that the use of the γ-GNG algorithm for unsupervisely discovering gesture templates in the time domain, enabled us to obtain the best reported performances (to our knowledge) on the OPPORTUNITY benchmark. Our approach reaches a $F1$-score close to 0.8, which is definitely, much better than previous benchmarks on the same dataset [i.e., $\approx$0.6 in (Satizábal et al. 2013) and $\approx$0.6 in (Sagha et al. 2011)].

Acknowledgments This work was funded by the Hasler Foundation. We thank our colleagues from the Wearable Computing Lab. ETH, for useful and interesting discussions.

References

Aggarwal JK, Ryoo MS (2011) Human activity analysis: a review. ACM Comput Surv 43(3):1–16. doi:10.1145/1922649.1922653 (ISSN 0360–0300)

Amft O, Tröster G (2008) Recognition of dietary activity events using on-body sensors. Artif Intell Med 42(2):121–136. doi:10.1016/j.artmed.2007.11.007 (ISSN 0933–3657)

Baños O, Damas M, Pomares H, Rojas I, Tóth MA, Amft O (2012) A benchmark dataset to evaluate sensor displacement in activity recognition. In: Proceedings of the 2012 ACM Conference on Ubiquitous Computing, UbiComp '12, New York, ACM, pp. 1026–1035. doi:10.1145/2370216.2370437 (ISBN 978-1-4503-1224-0)

Baños O, Damas M, Pomares H, Rojas I (2013) Activity recognition based on a multi-sensor meta-classifier. In: Rojas I, Joya G, Cabestany J (eds) Advances in computational Intelligence, volume 7903 of Lecture Notes in Computer Science. Springer, Heidelberg, pp. 208–215. doi:10.1007/978-3-642-38682-4_24 (ISBN 978-3-642-38681-7)

Bache K, Lichman M (2013) UCI machine learning repository. http://archive.ics.uci.edu/ml

Browne G, Berry E, Narinder K, Hodges S, Smyth G, Watson P, Wood K (2011) Sensecam improves memory for recent events and quality of life in a patient with memory retrieval difficulties. Memory 19(7):713–722

Bulling A, Blanke U, Schiele B (2014) A tutorial on human activity recognition using body-worn inertial sensors. ACM Comput Surv 46:1–33

Chavarriaga R, Bayati H, Del Millán J (2013) Unsupervised adaptation for acceleration-based activity recognition: Robustness to sensor displacement and rotation. Pers Ubiquitous Comput 17(3):479–490. doi:10.1007/s00779-011-0493-y (ISSN 1617–4909)

Chen L, Hoey J, Nugent CD, Cook DJ, Yu Z (2012) Sensor-based activity recognition. Syst Man Cybern Part C Appl Rev IEEE Trans 42(6):790–808. doi:10.1109/TSMCC.2012.2198883 (ISSN 1094–6977)

de Vries B, Principe JC (1992) The gamma model, a new neural model for temporal processing. Neural Netw 5(4):565–576. doi:10.1016/S0893-6080(05)80035-8 (ISSN 0893–6080)

Delachaux B, Rebetez J, Perez-Uribe A, Satizábal HF (2013) Indoor activity recognition by combining one-vs-all neural network classifiers exploiting wearable and depth sensors. In: Proceedings of the International Work-Conference. on Artificial Neural Networks. Springer, Heidelberg, pp 216–223

Estévez PA, Hernández R (2011) Gamma-filter self-organizing neural networks for time series analysis. In: Laaksonen J, Honkela T (eds) Advances in self-organizing maps, volume 6731 of Lecture Notes in Computer Science. Springer, Berlin Heidelberg, pp 151–159. doi:10.1007/978-3-642-21566-7_15 (ISBN 978-3-642-21565-0)

Fritzke B (1995) A growing neural gas network learns topologies. In: Tesauro G, Touretzky DS, Leen TK (eds) Advances in Neural Information Processing Systems 7. MIT Press, Cambridge, pp 625–632

Hartmann B, Link N (2010) Gesture recognition with inertial sensors and optimized dtw prototypes. In: Systems Man and Cybernetics (SMC), 2010 IEEE International Conferenc. pp 2102–2109. doi:10.1109/ICSMC.2010.5641703

Hondori HM, Khademi M, Lopes CV (2012) Monitoring intake gestures using sensor fusion (microsoft kinect and inertial sensors) for smart home tele-rehab setting. In: IEEE HIC 2012 Engineering in Medicine and Biology Society Conference on Healthcare Innovation.

Kasteren TLM, Englebienne G, Kröse BJA (2011) Human activity recognition from wireless sensor network data: Benchmark and software. In: Chen L, Nugent CD, Biswas J, Hoey J (eds) Activity recognition in pervasive intelligent environments, volume 4 of Atlantis Ambient and Pervasive Intelligence. Atlantis Press, pp. 165–186. doi:10.2991/978-94-91216-05-3_8 (ISBN 978-90-78677-42-0)

Kepski M, Kwolek B (2012) Fall detection on embedded platform using kinect and wireless accelerometer. In: Proceedings of the 13th International Conference on Computers Helping People with Special Needs— Volume Part II, ICCHP'12. Springer, Heidelberg, pp. 407–414. doi:10.1007/978-3-642-31534-3_60 (ISBN 978-3-642-31533-6)

Leppanen J, Eronen A (2013) Accelerometer-based activity recognition on a mobile phone using cepstral features and quantized gmms. In: Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference. pp 3487–3491. doi:10.1109/ICASSP.2013.6638306

Lukowicz P, Pirkl G, Bannach D, Wagner F, Calatroni A, Förster K, Holleczek T, Rossi M, Roggen D, Tröster G, Doppler J, Holzmann C, Riener A, Ferscha A, Chavarriaga R (2010) Recording a complex, multi modal activity data set for context recognition. In: ARCS Workshops, pp 161–166

Nguyen-Dinh L-V , Roggen D, Calatroni A, Troster G (2012) Improving online gesture recognition with template matching methods in accelerometer data. In: Intelligent Systems Design and Applications (ISDA), 2012 12th International Conference on, pp 831–836. doi:10.1109/ISDA.2012.6416645

Ni B, Wang G, Moulin P (2013) Rgbd-hudaact: A color-depth video database for human daily activity recognition. In: Fossati A, Gall J, Grabner H, Ren X, Konolige K (eds) Consumer depth cameras for computer vision, Advances in computer vision and pattern recognition. Springer, London, pp 193–208. doi:10.1007/978-1-4471-4640-7_10 (ISBN 978-1-4471-4639-1)

Lara ÓD, Pérez AJ, Labrador MA, Posada JD (2012) Centinela: a human activity recognition system based on acceleration and vital sign data. Pervasive Mobile Comput 8(5):717–729. doi:10.1016/j.pmcj.2011.06.004 (ISSN 1574–1192)

Principe JC, de Vries B, de Oliveira PG (1993) The gamma-filter-a new class of adaptive iir filters with restricted feedback. Trans Sig Proc 41(2):649–656. doi:10.1109/78.193206 (ISSN 1053–587X)

Rawassizadeh R, Tomitsch M, Wac K, Min Tjoa A (2013) Ubiqlog: a generic mobile phone-based life-log framework. Personal and Ubiquitous Computing, 17(4):621–637. doi:10.1007/s00779-012-0511-8 (ISSN 1617–4909)

Rebetez J, Satizábal HF, Perez-Uribe A (2013) Reducing user intervention in incremental activityrecognition for assistive technologies. In: Proceedings of the 2013 International Symposium on Wearable Computers, ISWC '13. New York. ACM. pp 29–32. doi:10.1145/2493988.2494350 (ISBN 978-1-4503-2127-3)

Van Rijsbergen CJ (1979) Information Retrieval. 2nd edn. Butterworth-Heinemann, Newton (ISBN 0408709294)

Roggen D, Calatroni A, Rossi M, Holleczek T, Förster K, Tröster G, Lukowicz P, Bannach D, Pirkl G, Ferscha A, Doppler J, Holzmann C, Kurz M, Holl G, Chavarriaga R, Sagha H, Bayati H, Creatura M, Millán JR (2010) Collecting complex activity datasets in highly rich networked sensor environments. In: Networked Sensing Systems (INSS), 2010 Seventh International Conference on, pp 233–240. doi:10.1109/INSS.2010.5573462

Sagha H, Digumarti ST, Millán JR, Chavarriaga R, Calatroni A, Roggen D, Tröster G (2011) Benchmarking classification techniques using the Opportunity human activity dataset. In: 2011 Ieee International Conference On Systems, Man, And Cybernetics (Smc), IEEE International Conference on Systems Man and Cybernetics Conference Proceedings. IEEE service center, Piscataway, pp 36–40

Satizábal HF, Rebetez J , Perez-Uribe A (2013) Semi-supervised discovery of time-series templates for gesture spotting in activity recognition. In: Proceedings of the 2nd International Conference in Pattern Recognition Applications and Methods. pp 573–576

Shotton J, Sharp T, Kipman A, Fitzgibbon A, Finocchio M, Blake A, Cook M, Moore R (2013) Real-time human pose recognition in parts from single depth images. Commun ACM 56(1):116–124. doi:10.1145/2398356.2398381 (ISSN 0001–0782)

Stiefmeier T, Roggen D, Ogris G, Lukowicz P, Lukowicz P (2008) Wearable activity tracking in car manufacturing. Pervasive Comput IEEE 7(2):42–50. doi:10.1109/MPRV.2008.40 (ISSN 1536–1268)

van Kasteren TLM, Englebienne G, Kröse BJA (2011) Hierarchical activity recognition using automatically clustered actions. In: Proceedings of the Second international conference on Ambient Intelligence, Am I'11. Springer, Heidelberg, pp 82–91. doi:10.1007/978-3-642-25167-2_9 (ISBN 978-3-642-25166-5)

Xue Y, Jin L (2010) A naturalistic 3d acceleration-based activity dataset amp; benchmark evaluations. In: Systems Man and Cybernetics (SMC), 2010 IEEE International Conference. pp 4081–4085. doi:10.1109/ICSMC.2010.5641790