

GestureKeeper: Gesture Recognition for Controlling Devices in IoT Environments

Vasileios Sideridis*, Andrew Zacharakis*[†], George Tzagkarakis*, and Maria Papadopouli*[†]

*Institute of Computer Science, Foundation for Research and Technology-Hellas, Heraklion, Greece

[†]Department of Computer Science, University of Crete, Heraklion, Greece

Abstract—This paper introduces and evaluates the GestureKeeper, a robust hand-gesture recognition system based on a wearable inertial measurements unit (IMU). The identification of the time windows where the gestures occur, without relying on an explicit user action or a special gesture marker, is a very challenging task. To address this problem, GestureKeeper identifies the start of a gesture by exploiting the underlying dynamics of the associated time series using a recurrence quantification analysis (RQA). RQA is a powerful method for nonlinear time-series analysis, which enables the detection of critical transitions in the system’s dynamical behavior. Most importantly, it does not make any assumption about the underlying distribution or model that governs the data. Having estimated the gesture window, a support vector machine is employed to recognize the specific gesture. Our proposed method is evaluated by means of a small-scale pilot study at FORTH and demonstrated that GestureKeeper can identify correctly the start of a gesture with a 87% mean balanced accuracy and classify correctly the specific hand-gesture with a mean accuracy of over 96%. To the best of our knowledge, GestureKeeper is the first automatic hand-gesture identification system based only on accelerometer. The performance analysis reveals the predictive power of the features and the system’s robustness in the presence of additive noise. We also performed a sensitivity analysis to examine the impact of various parameters and a comparative analysis of different classifiers (SVM, random forests). Most importantly, the system can be extended to incorporate a large dictionary of gestures and operate without further calibration for a new user.

Index Terms—Hand-gesture identification and recognition, inertial measurement unit, support vector machine, recurrence quantification analysis

I. INTRODUCTION

In general, hand-gesture recognition systems can be classified in two categories, according to the type of sensors they employ, namely, the camera- and the wearable-based ones. The camera-based systems can achieve high recognition accuracy, but at a relatively high computational cost [1]. The performance of these systems is sensitive in the background, light conditions, and room geometry, and constrained by the field of view of the camera. On the other hand, sensor-based systems, worn on the wrist, leg, arm, chest, ankle, head and/or waist, employ accelerometers, gyroscopes and magnetometers, barometers, body sensor networks [2], electromyography sensors [3], or even sound sensors [4]. They

This work is partially funded by the “Human Resources Development, Education and Life Lifelong Learning” for the implementation of the European Social Fund and the Youth Employment Initiative, and the Hellenic Foundation for Research and Innovation (HFRI) and the General Secretariat for Research and Technology (GSRT) under grant agreement No. 2285 (neuronXnet). Contact author Maria Papadopouli.

have relatively small cost, are not particularly sensitive to the environmental conditions (e.g. light or geometry conditions), and can function in indoor and outdoor spaces. Although, in general, wearable sensors are energy-constrained, due to the rapid advances of the micro-electromechanical technologies in reducing their size and enhancing their energy efficiency, they have gained a lot of attention in hand-gesture recognition (HGR), human activity recognition (HAR) [5]–[9] and even in human writing recognition [10]. These systems collect data from the on-board sensors and apply either machine learning algorithms [11]–[16], mathematical models [4], [17], [18], fuzzy control techniques [19] or simple threshold-based algorithms [20] to recognize the user’s gestures. Inertial measurement units are extremely useful and commonly-used for orientation/heading estimation [21], [22]. The prevalence of accelerometers, gyroscopes and magnetometers in smartphones for estimating their orientation has enabled various interesting applications in the gesture recognition domain.

This paper introduces and evaluates the GestureKeeper, an innovative robust hand-gesture identification and recognition system based on a wearable inertial measurements unit (IMU). In the context of daily activities, the user can control an appliance by performing a specific gesture. The identification of starting and ending points of the time windows, where the gestures occur, without relying on an explicit user action, as in [11], or a special gesture marker, as in [23], is challenging. To address this problem, GestureKeeper identifies the start of a gesture by exploiting the underlying dynamics of the associated time series using recurrence quantification analysis (RQA). RQA is a powerful method for nonlinear time-series analysis, which enables the detection of critical transitions in the system’s dynamics (e.g. deterministic, stochastic). Most importantly, it does *not* make any assumption about the underlying distribution or model that governs the data. Moreover, it can be used even for relatively small and non-stationary datasets. More specifically, our proposed method capitalizes on the efficiency of RQA to extract the underlying dynamics of a recorded sensor data stream by mapping the associated time series in a higher-dimensional phase space of trajectories. A major advantage of RQA is its *fully self-tuned nature*, in the sense that no prior parameter fine-tuning is required in a manual fashion.

To the best of our knowledge, GestureKeeper is the *first automatic hand-gesture identification system based only on accelerometers*. Furthermore, it can recognize accurately a

dictionary of 12 hand-gestures by applying support vector machines (SVM) on a hybrid set of statistical and sample-type features. The evaluation was performed in a small-scale pilot study at FORTH. This paper demonstrates that GestureKeeper can recognize gestures from our 12-hand-gesture dictionary with a mean accuracy of about 96%. The analysis also reveals the predictive power of the features and the system's robustness in the presence of additive noise. We performed a sensitivity analysis to examine the impact of various parameters. Finally, to comparatively assess the performance of SVM, we also applied random forests. SVM still achieves a higher accuracy. The rest of the paper is organized as follows: Section II overviews our system, its architecture, and the two main sub-systems, while Section III summarizes our main conclusions and future research directions.

II. GESTUREKEEPER SYSTEM DESIGN

GestureKeeper consists of a wearable sensor and a server. The wearable sensor sends periodically its collected measurements to the server, while the server performs the gesture identification and recognition. Our dictionary consists of 12 gestures. Their names, short descriptions, and trajectories in space are shown below.

- **Up:** Vertical movement towards the ceiling
- **Down:** Vertical movement towards the ground
- **Left:** Horizontal movement to the left
- **Right:** Horizontal movement to the right
- **CW:** Clock-wise rotational movement
- **CCW:** Counter Clock-wise rotational movement
- **Z:** Z trajectory starting from above
- **AZ:** Mirror Z trajectory starting from below
- **S:** Wave trajectory to the right
- **AS:** Wave trajectory to the left
- **Push:** Horizontal movement away from the body
- **Pull:** Horizontal movement towards the body

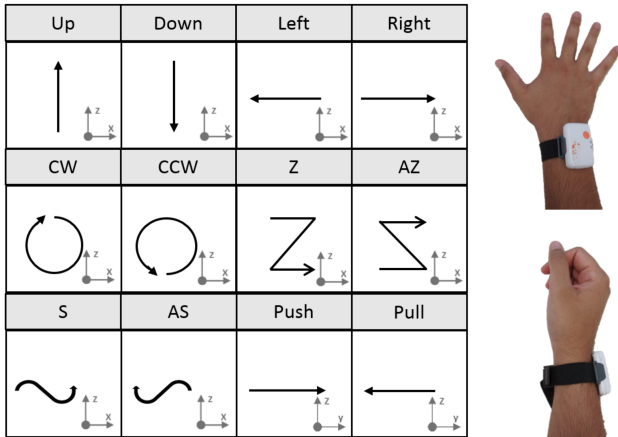


Fig. 1. Dictionary of gestures and hand placement.

Experiments and Data Collection. We employed the shimmer3 [24], a wearable device equipped with 3-axis accelerometer, gyroscope, and magnetometer. We first calibrated the sensors and then configured each of them according to the needs of our implementation. For example, due to the relatively low expected acceleration, we chose a relatively

small detection range for the accelerometer increasing this way its resolution. The final sampling frequency was set to 50 Hz, which is considered sufficient (given our observations from a number of fast steep movement experiments). An additional increase of the frequency will only increase the power consumption of the device and the measurement size without, however, enhancing the amount of information about the user's movement. Finally, the data streaming and logging applications are responsible to collect the data during the experiments.

For the recognition sub-system, we performed a small field study with 15 subjects (9 female, 6 male). Each subject repeated a number of gestures (from the predefined dictionary of gestures), collecting in total 900 different repetitions. For each repetition, 63 statistical features and 30 acceleration-type features, which represent values of acceleration in the x, y and z axes, were extracted. This dataset had the isolated periods during which a gesture was performed. For the identification sub-system, a *new* dataset was produced, containing gestures as well as activities of daily living (ADL). The new dataset has a total duration of 3 hours and 45 minutes, with measurements collected from 4 new subjects (2 female, 2 male).

A. Gesture Identification

We first focus on the problem of identifying the start of a gesture in a recorded data stream. We speculate that different gestures are characterized by distinct dynamics of the associated time series. This motivated the use of RQA, which enables the detection of transitions in the dynamical behavior (e.g. deterministic, chaotic, etc.) of the observed system. A major advantage of RQA is its *fully self-tuned nature*, in the sense that no prior parameter fine-tuning is required in a manual fashion. More specifically, a recurrence plot (RP) is derived first, which depicts those times at which a state of a dynamical system recurs, thus revealing all the times when the *phase space trajectory* of the dynamical system visits roughly the same area in the phase space. To this end, RPs enable the investigation of an m -dimensional phase space trajectory through a two-dimensional representation of its recurrences. Such recurrence of a state occurring at time i , at a different time j is represented within a square matrix with ones (recurrence) and zeros (non-recurrence), where both axes are time axes. Given a time series of length N , $\{r_i\}_{i=1}^N$, a phase space trajectory is reconstructed via time-delay embedding,

$$\mathbf{x}_i = [r_i, r_{i+\tau}, \dots, r_{i+(m-1)\tau}] , \quad i = 1, \dots, N_s , \quad (1)$$

where m is the embedding dimension, τ is the delay, and $N_s = N - (m-1)\tau$ is the number of states. Having constructed a phase space representation, an RP is defined as follows,

$$\mathbf{R}_{i,j} = \Theta(\varepsilon - \|\mathbf{x}_i - \mathbf{x}_j\|_p) , \quad i, j = 1, \dots, N_s , \quad (2)$$

where $\mathbf{x}_i, \mathbf{x}_j \in \mathbb{R}^m$ are the states, ε is a threshold, $\|\cdot\|_p$ denotes a general ℓ_p norm, and $\Theta(\cdot)$ is the Heaviside step function, whose discrete form is defined by

$$\Theta(n) = \begin{cases} 1, & \text{if } n \geq 0 \\ 0, & \text{if } n < 0 \end{cases} , \quad n \in \mathbb{R} . \quad (3)$$

Typically, several linear (and/or curvilinear) structures appear in RPs, which provide hints about the time evolution of the high-dimensional phase space trajectories. Besides, a major advantage of RPs is that they can be applied to *short* and *non-stationary* data. The visual interpretation of RPs, which is often difficult and subjective, is enhanced by means of several numerical measures for the quantification of the structure and complexity of RPs [25]. These quantification measures provide a global picture of the underlying dynamical behavior during the entire period covered by the recorded data. The temporal evolution of RQA measures and the subsequent detection of transient dynamics are enabled for each recorded sensor stream by employing a windowed version of RQA. Doing so, the corresponding quantification measures are computed in small windows, which are then merged to form our feature matrix. Furthermore, it is noted that the length of the sliding window yields a compromise between resolving small-scale local fluctuations and detecting more global recurrence structures.

The gesture identification of the GestureKeeper employs two RQA metrics, namely, the recurrence rate (RR) and the transitivity (TRA) ([26]), obtained from the y-axis acceleration data, in order to form the feature matrix¹.

Estimation of Embedding Parameters. In our implementation, the optimal time delay τ is estimated as the first minimum of the average mutual information (AMI) function [27]. Concerning the embedding dimension m , a minimal sufficient value is estimated using the method of false nearest neighbours (FNN) [28]. Furthermore, the Euclidean norm is used as our selected distance metric for the construction of the RP, while a rule-of-thumb is currently used to set the threshold $\epsilon = 0.2\sqrt{m}$. The window length refers to the size of the signal in which RQA is performed each time before being shifted by a step size to the next values. We selected a window size of 125 samples, which represents approximately 2.5 seconds of information, sufficient for capturing even the longest of the dictionary's gestures. The step was set equal to 25 samples (i.e., 80% overlap between consecutive windows). By applying the above criteria on our data, the estimated embedding parameters are equal to $m = 4$ and $\tau = 1$. Although the empirical rule for selecting ϵ yields $\epsilon = 0.4$, a higher accuracy was observed for $\epsilon = 0.1$.

An SVM classifier based on these two features was then employed to classify the data in two classes, namely, gestures and ADL, thus distinguishing the gestures from the rest of the hand movements.

B. Gesture Recognition

The gesture recognition is based on an SVM classifier using the radial kernel². For the classification, we employed two types of features, namely the *statistical features* and *samples of the acceleration signal*. The statistical features include the mean, median, root mean square (RMS), standard deviation,

variance, skewness and kurtosis, of the 3D acceleration, angular velocity, and magnetism time series provided by the sensor. The sample based features are formed by a re-sampling process of the x, y and z axis acceleration³. After re-sampling, the new signal is composed of a fixed number of samples for each acceleration time series. The final set of features includes the statistical ones (introduced earlier) along with this number of samples of the re-sampled acceleration signal for each of the x-, y-, and z-axis time series.

C. Performance Analysis

Gesture Identification: GestureKeeper first utilizes RQA in order to extract features for the identification sub-system, thereafter uses an SVM model, trained with the aforementioned features, for identifying correctly the windows that contain gestures. Note that the ADL and gesture classes are highly unbalanced: only 0.5% of the data belong to the class “gestures” and the 99.5% is ADL. To analyze the performance of the identification process, we trained the SVM classifier (with a polynomial kernel of degree = 3, coefficient = 2, cost = 3, and gamma = 0.95⁴) using all subjects except one, which was then used for testing. Given that the ADL and gesture classes are highly unbalanced, we randomly selected a subset of the ADL class of equal size as the gesture one. We performed the training and testing on this dataset. This process was repeated for 100 iterations. We reported the mean accuracy for each subject of the testing. The accuracy for each subject varies from 76.9% to 91.1% (with a mean accuracy of 87.21%).

TABLE I
LIST OF ALL THE PARAMETERS THAT WERE USED IN GESTUREKEEPER

RQA		SVM Identification		SVM Recognition	
Parameters	Value	Parameters	Value	Parameters	Value
Distance Metric	Eucl. Norm	Kernel	polyn.	Kernel	radial
Window size	125	Gamma (γ)	0.95	Gamma (γ)	0.005
Window step	25	Cost (c)	3	Cost (c)	1
Delay (τ)	1	Degree	3		
Dimensions (m)	4	Coefficient	2		
Threshold (ϵ)	0.1				

Gesture Recognition: As mentioned earlier in Sec. II-B, an SVM model with different parameters is employed for the final classification of the gestures in one of the 12 in total classes of our dictionary. To assess the predictive power of the statistical features, we performed the following process: First, we permuted a feature, keeping the values of the remaining features of the dataset fixed. Then, the model is trained and tested with this dataset. The mean accuracy is based on 100 repetitions. We then repeated the same process, selecting each time to permute a different feature from the original dataset. The multiclass classification model employs the “one-against-one” approach, in which $12 \cdot (12 - 1) / 2$ binary classifiers are trained and the appropriate class is selected by a voting scheme. The

¹We initially employed an extended RQA measures set but observed that the aforementioned two measures are sufficient to identify the gesture time-windows.

²Radial kernel was shown in our sensitivity analysis to have the best accuracy among the polynomial, linear, and sigmoid.

³The original acceleration signal consists of 3 time series, one for each axis, of unknown length (since it depends on the particular gesture and the user who performed the movement).

⁴We also examined various classifiers using different kernels, such as linear, sigmoid, radial, and parameters values. The reported results were obtained using the above values.

mean accuracy for the different permuted features is shown in Fig. 2. The horizontal line indicates the mean accuracy for the original dataset without any feature permutation. The larger the decrease in the accuracy, the more significant the information that the corresponding feature provides in gesture recognition. It appears that the mean and skewness of the z-axis angular velocity are the most significant ones.

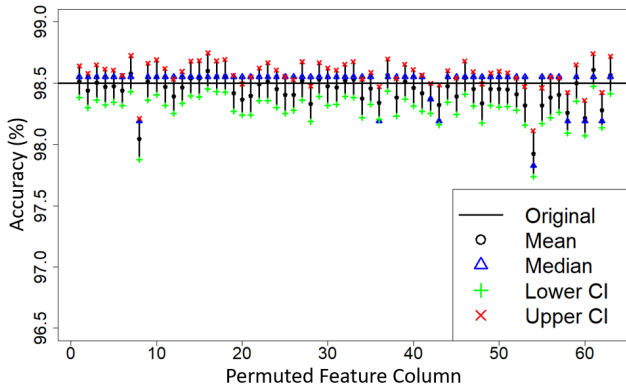


Fig. 2. Mean accuracy under a single-feature permutation (indicated in the x-axis). Features are based on simple statistics of data collected from accelerometer, gyroscope, and magnetometer from each of the three dimensions. Original line: no feature has been permuted. (CI = 95% confidence interval)

A similar procedure was also applied for the acceleration-sample-based features. We examined a different number of samples for each time series (in the range of 4-12). In the case of ten samples per time series, the fourth, fifth, sixth value of the re-sampled x-axis acceleration have a significant impact on accuracy, while the remaining features exhibit a similar predictive power for the gesture recognition (as shown in Fig. 3). However, unlike the statistical features, all the acceleration-sample-based ones have a substantial contribution on the accuracy (as the accuracy using the original dataset is greater from the one using the permuted ones). Note that the first 10 samples that refer to acceleration in x axis carry more information than the others. Specifically, the most important feature tends to be the middle value of the x-axis signal (sample 5). Fig. 4 shows the accuracy of the gesture classification, when only the sample features are used, as a function of the sample size. The impact on the accuracy is prominent for a sample size of 5 or less.

Our gesture recognition classifier employs in total 73 features, namely the 43 most significant statistical features and 10 samples-based features from each acceleration time series. The above analysis was performed using the default SVM hyper-parameters, namely, cost $c = 1$ and $\gamma = 1/\text{number of features}$.

Experiments and SVM Tuning. We train the gesture recognition classifier using the data collected from the 14 out of 15 subjects for training and the other one's for testing. We performed tests for all the 15 combinations of training and testing partitions.

The hyper-parameters that we tuned are the cost (c) and gamma (γ) values. The cost represents the weight for penalizing the “soft margin”. Consequently, a large cost value

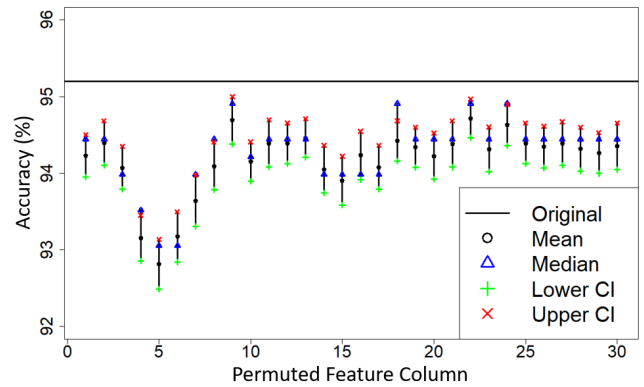


Fig. 3. Mean accuracy under a single-feature permutation (indicated in the x-axis). The first ten features correspond to a re-sampling of the acceleration data in the x-axis, while the 11th-20th (21th-30th) features correspond to y-axis (z-axis), respectively. Original line: no feature has been permuted. (CI = 95% confidence interval)

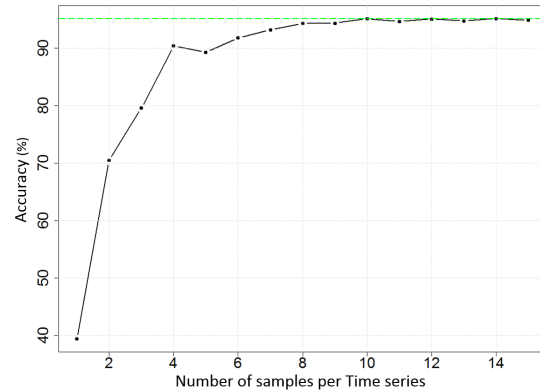


Fig. 4. Mean accuracy vs. number of samples per time series.

penalizes the SVM for data points within the margin or on the wrong side of the dividing hyper-plane. For this reason, for a large c , the SVM will try to find more complex hyper-planes, and possibly smaller margins that leave less data points in the wrong side, but is also more prone to over-fitting. In contrast, for small c values, the margins of the SVM will be larger and the hyper-plane less complex, making it more robust but also less accurate. This motivates the need for carefully addressing the tradeoff of accuracy and robustness. The gamma parameter refers to the kernel and depends on the number of features that each specific implementation has. Intuitively, gamma controls the number of support vectors and by extension the sensitivity of the decision boundary (hyper-plane), which will affect whether or not some data points near the margins will be ignored. A sensitivity analysis with different cost and gamma values reports the best performance for $c = 1, \gamma = 0.005$. Fig. 5 shows the accuracy of each subject's data as testing set for the following datasets, namely, all statistical features, only significant statistical ones, only samples, both significant statistical features and the samples (proposed dataset), with a mean accuracy of 94.44%, 94.44%, 92.88%, and 96.22%,

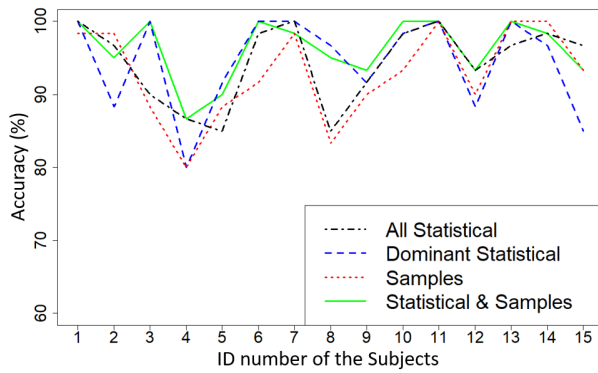


Fig. 5. Mean accuracy of each subject's data for different datasets.

respectively. The best performance is obtained when all the significant statistical features and samples-based ones are employed. The presence of all the parameters that were used in this paper is considered essential for any further experimental evaluation. Therefore Table I shows the parameters for the RQA and the SVM model of the first and second sub-system, respectively.

To comparatively evaluate the recognition sub-system, we also developed classifiers based on random forests. We evaluated them using the (first) dataset (that was also used for the recognition sub-system based on SVM). After training and tuning⁵, random forests reported a best accuracy of 88% as opposed to 96% of the proposed method using SVM.

Finally, we assessed the impact of noise on the accuracy of the gesture recognition sub-system. The mean accuracy was increased to 97.89%, when each feature vector of the original (clear) training dataset was augmented by a copy of a corrupted one (produced by adding Gaussian noise with standard deviation of 0.5).

III. CONCLUSION

This paper presents GestureKeeper which employs an accelerometer, gyroscope and magnetometer, from a wearable IMU, to first identify time-windows that contain a gesture, and then, recognize which specific gesture it is. GestureKeeper uses features based on statistical properties and acceleration samples. It can accurately recognize gestures from our 12-hand-gesture dictionary, exhibiting its best performance when the combination of features are used (e.g., about 96% mean accuracy). With the noise addition and feature selection, the mean accuracy is increased to over 97%. It is modular and can be extended to recognize a larger gesture dictionary size. The pilot field study was performed in a relatively controlled small-scale environment. We plan to extend the evaluation under more realistic conditions. Moreover, it is critical to correctly center the gesture for each of the identified time-windows,

⁵The sensitivity analysis for the random forests reported best accuracy for 100 trees in the forest and a maximum depth of the tree equal to 10. The minimum number of samples required to split an internal node and the minimum number of samples required to be at a leaf node were left with the default values of 2 and 1 respectively as they did not seem to effect the accuracy.

as we observed a significant accuracy drop. We will explore the use of long-short term memory (LSTM) networks and conditional random fields in the gesture recognition to address these challenges.

REFERENCES

- [1] F. Dominio *et al.*, "Hand gesture recognition with depth data," *ACM/IEEE Int' Work. on Anal. & Retrieval of Tracked Events & Motion in Imagery Stream*, 2013.
- [2] V. Jaijongrak, S. Chantasuban, and S. Thiemjarus, "Towards a BSN-based gesture interface for intelligent home applications," *ICCCAS-SICE*, 2009.
- [3] V. E. Kosmidou and L. J. Hadjileontiadis, "Intrinsic mode entropy: An enhanced classification means for automated Greek Sign Language gesture recognition," *Int' Conf. of IEEE Engin. in Medicine & Biology Society*, 2008.
- [4] H. Basanta, Y. Huang and T. Lee, "Assistive design for elderly living ambient using voice and gesture recognition system," *IEEE Int' Conf. on Systems, Man & Cybernetics*, 2017.
- [5] O. A. B. Penatti and M. F. S. Santos, "Human activity recognition from mobile inertial sensors using recurrence plots," *Published in ArXiv*, 2017.
- [6] E. Garcia-Ceja, Md. Z. Uddin, and J. Torresen, "Classification of recurrence plots distance matrices with a convolutional neural network for activity recognition," *Procedia Comp. Science*, **130**, 2018.
- [7] A. Bulling, U. Blanke, and B. Schiele, "A tutorial on human activity recognition using body-worn inertial sensors," *ACM Comput. Surv.*, **46**(3), Art. 33, 2014.
- [8] S. Savvakis, G. Tsagkatakis, A. Panousopoulou and P. Tsakalides, "Matrix and tensor completion on a human activity recognition framework," *IEEE J. Biomedical and Health Informatics*, **21**(6), 2017.
- [9] N. Twomey *et al.*, "A comprehensive study of activity recognition using accelerometers," *Informatics* 5, 2018.
- [10] S. Agrawal *et al.*, "Using mobile phones to write in air," *ACM Mobisys*, 2011.
- [11] J. Ducloux *et al.*, "Accelerometer-based hand gesture recognition system for interaction in digital TV," *IEEE Int' Instrum. & Measur. Tech. Conf.*, 2014.
- [12] Y. Ma *et al.*, "Hand gesture recognition with convolutional neural networks for the multimodal UAV control," *Work. on Research, Education and Development of Unmanned Aerial Systems*, 2017.
- [13] F. Alemuda and F. J. Lin, "Gesture-based control in a smart home environment," *IEEE Int' iThings & GreenCom & CPSCom & SmartData*, 2017.
- [14] C. Zhu, Q. Cheng, and W. Sheng, "Human intention recognition in smart assisted living systems using a hierarchical hidden Markov model," *IEEE Int' Conf. on Automation Science & Engineering*, 2008.
- [15] T. Tai *et al.*, "Sensor-based continuous hand gesture recognition by long short-term memory," *IEEE Sen. Lett.*, **2**(3), 2018.
- [16] L. Xu, W. Yang, Y. Cao, and Q. Li, "Human activity recognition based on random forests," *Int' Conf. on Natural Comp., Fuzzy Systems & Knowledge Discov.*, 2017.
- [17] D. Zhang, X. Wu, and C. Wang, "Fine-grained and real-time gesture recognition by using IMU sensors," *IEEE Int' Conf. on Parallel & Distributed Systems*, 2017.
- [18] T. Do *et al.*, "Design of an infrastructureless in-door localization device using an IMU sensor," *IEEE Int' Conf. on Robot. & Biomim.*, 2015.
- [19] H. Chu, S. Huang, and J. Liaw, "An acceleration feature-based gesture recognition system," *IEEE Int' Conf. on Systems, Man & Cybernetics*, 2013.
- [20] H. Yun and B. Song, "Dynamic characteristic analysis of users' motions for human smartphone interface," *Int' Conf. on Computing & Networking Tech.*, 2012.
- [21] S. O. H. Madgwick, A. J. L. Harrison, and R. Vaidyanathan, "Estimation of IMU & MARG orientation using a gradient descent algorithm," *IEEE Intl. Conf. on Rehabilitation Robotics*, 2011.
- [22] C. Jagadish and B.-C. Chang, "Diversified redundancy in the measurement of Euler angles using accelerometers and magnetometers," *IEEE Conf. on Decision and Control*, 2007.
- [23] D. Lee, H. Yoon, and J. Kim, "Continuous gesture recognition by using gesture spotting," *16th ICCAS*, 2016.
- [24] Shimmer products, <https://bit.ly/2IpYrZP>.
- [25] J. Zbilut and C. Webber, "Embeddings and delays as derived from quantification of recurrence plots," *Phys. Lett. A*, **171**(34), 1992.
- [26] N. Marwan, M. Romano, M. Thiel, and J. Kurths, "Recurrence plots for the analysis of complex systems," *Phys. Rep.*, **438**(5-6), 2007.
- [27] A. Fraser and H. Swinney, "Independent coordinates for strange attractors from mutual information," *Phys. Rev. A*, **33**, 1986.
- [28] M. B. Kennel *et al.*, "Determining embedding dimension for phase-space reconstruction using a geometrical construction," *Phys. Rev. A*, **45**, 1992.