S & M 1882

Human Activity Recognition Based on Smart Chair

Chien-Cheng Lee,* Lamin Saidy, and Fitri

Department of Electrical Engineering, Yuan Ze University,
No. 135, Yuandong Rd, Zhongli District, Taoyuan City, 320, Taiwan

(Received April 16, 2018; accepted March 20, 2019)

Keywords: smart chair, pressure sensor, random forest, extremely randomized tree

We present a smart chair that can detect and classify some common daily activities of elderly people. The chair has the potential to be a huge source of information on the behaviors of people since most indoor activities are performed in sedentary positions. The proposed smart chair comprises six pressure sensors mounted in a chair, together with a Raspberry Pi to collect raw data. The mounted pressure sensors collect signals and transmit them to a server for processing and analysis while the user sits in the chair. Five different activities are detected and classified by these sensors: working at the desk, eating, napping, coughing, and watching TV. In an effort to achieve the best classification of these activities, three different machine learning algorithms are employed and their accuracy scores were compared. These algorithms are the random forest (RF), extremely randomized trees (ERTs), and support vector machine (SVM). The experimental results have proven the ERT to be the best classifier in this survey, since it yielded a classification accuracy above 98% over the testing data.

1. Introduction

Activity monitoring systems are paramount in giving care to individuals, especially to elderly people.(1) The activities of humans, regardless of their age group, play a huge role in determining the kind of therapy they needed to undergo. Therefore, building systems to recognize activities performed by humans can improve the health-care services and the conditions of individuals as they age.(2,3)

Camera-based monitoring systems are the most popular activity monitoring systems in the modern world. Although camera surveillance systems are effective in keeping track of human activities, they do not offer privacy to individuals. Wearable devices can also be used to monitor daily activities.(4,5) Thus, several companies provide wearable solutions to protect individual privacy. Nevertheless, these devices usually work on rechargeable batteries, which need periodic recharging. This makes the wearable devices inconvenient for daily use.

A considerable amount of research has been carried out to develop various technologies for recognizing the activities performed by elderly people. As proposed by some researchers, monitoring the activities performed by elderly people at home offers a solution in giving them

*Corresponding author: e-mail: cclee@saturn.yzu.edu.tw

https://doi.org/10.18494/SAM.2019.2280

indirect care.(1,6) In the olden days, different sensing modalities were adopted to sense the activity of individuals. One such approach involved sensors and cameras.(1,7,8) Kang *et al.* developed an automatic human movement classification system for the aged, using a single waist-mounted triaxial accelerometer.(6) The system classifies the daily activities of aged individuals, such as sitting, lying down, standing, walking, running, falling, from sitting to standing, from standing to sitting, and moving from standing to lying down, from lying down to standing, from sitting to lying down, and from lying down to sitting. An algorithm of the hierarchical binary tree is used to classify these activities. In the same regard, Zouba *et al.* proposed a multisensory activity recognition approach in which video cameras and environmental sensors are used to recognize activities of interest performed by aged individuals at home.(8) The same authors used a similar approach to performing behavioral analysis of aged individuals using data from sensors.(9)

As privacy issues pose a major concern for activity-recognition camera systems, some researchers have turned their attention to static-posture detection instead. Static posture is held for a certain time and the physical exertion of maintaining the same posture or position is determined. Tan *et al.* reported a method in which an office chair is used to elucidate the occupant's actions and needs.(10) They mounted sensors on the seat cushion and the backrest of a chair to detect sixteen activities. Principal component analysis was carried out to classify the activities. Along the same line, Fu and Macleod presented a system for predicting the activities of individual subjects by feeding posture information to two classifiers, one for back posture and the other for leg posture.(11) The system has many potential applications, such as the analysis of subjects sitting or lying down, motion tracking in rehabilitation, interaction assistance, and the detection of anomalous activities. All the sensors they used were mounted on the seat cushion and the back of a chair. In another work, Kumar *et al.* proposed a Care–Chair system to classify nineteen fine-grained and complex sedentary user activities.(12) The system detects static postures and movements. The activities include napping, sitting still, looking back to the left, looking back to the right, tilting the head from side to side, nodding the head up and down, waving a hand, talking, sneezing, coughing, drinking, eating, hiccupping, crying, laughing, shouting, weeping, yawning, and yelling.

There is a need to devise a technology to retrieve information about people while they are seated. Such information helps determine and improve the health of not only elderly people but also individuals with poor health. Elderly people usually spend their time at home performing their activities in sedentary positions. In the previous studies, the common practice was to embed sensors in a chair to detect activities performed by individuals in a sedentary position. The number of detected activities depends on the number of sensors and their mounting positions. However, detecting too many activities may lead to great confusion in the system, especially in terms of detecting and classifying similar activities.

The aim of this work is to develop a low-cost, reliable smart chair to monitor and classify daily activities performed by elderly people at home. Thus, five common activities, that is, working at a desk, eating, napping, coughing, and watching TV, are detected using the proposed system. The proposed smart chair will facilitate the monitoring of aged people without human intervention.

2. Materials and Methods

2.1 System prototype design

The proposed smart chair is built from simple and sophisticated devices yet robust enough to detect five activities of individuals in sedentary positions. These activities include eating, working at the desk, watching TV, napping, and coughing. The smart chair is composed of six pressure sensors, an analog-to-digital (A/D) converter (MCP 3008), Raspberry Pi, and a conventional office chair. Four pressure sensors are placed on the seat to collect data while the subject is sitting upright and two on the backrest to collect information when the subject leans back. These sensors generate analog signals, and the A/D converter (MCP 3008) is used to convert the collected signals to digital signals for the Raspberry PI device. Raspberry Pi transmits the digital signals to a server for preprocessing and analysis. The setup is illustrated in Fig. 1.

The adopted Raspberry Pi 2 Model B is equipped with 512 MB SDRAM, 128 GB micro-SD storage, 40 GPIO, and four USB 2.0 ports. Force-sensitive resistors (FSRs) are employed as pressure sensors to collect raw signals. These FSR sensors allow the measurement of static and dynamic forces applied to the contact surface.[13] They are available in two types: circular FSR sensors and square FSR sensors. The square FSR sensors are used in this study because of their wider force sensitivity range of ~3 g to ~3 kg and are optimized for use in human touch control applications. The MCP 3008 device is a microchip with a successive approximation 10-bit A/D converter. It is a programmable device that provides four pseudo-differential input pairs. Communication with the devices is accomplished using a simple serial interface compatible with the SPI protocol. It is mainly employed in this research to convert analog signals read by the FSR sensors to digital signals for Raspberry Pi.

Through the A/D converter, Raspberry Pi collects the digital signals from FSR sensors attached to the chair. While the subject is performing activities, it stores the signals in CSV files and transmits them to the server for analysis. These devices are depicted in Fig. 2.

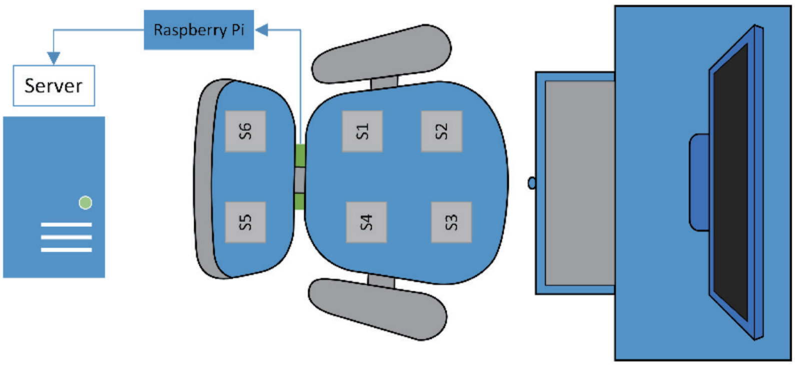


Fig. 1. (Color online) System architecture.

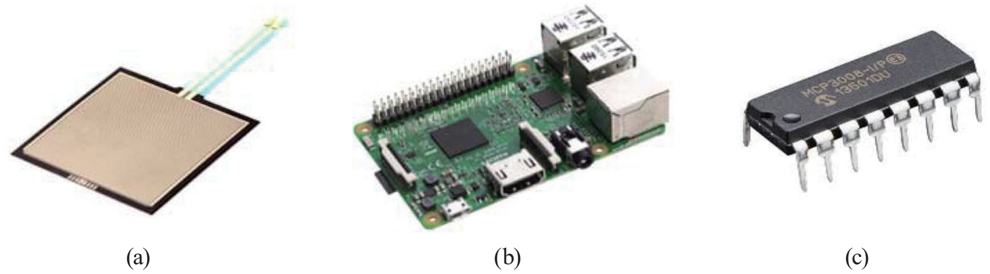


Fig. 2. (Color online) System devices. (a) Pressure sensor, (b) Raspberry Pi, and (c) MCP 3008 ADC.

2.2 Feature extraction

Feature extraction is an essential signal processing step prior to applying a learning algorithm. In an effort to improve the accuracy of the machine learning models, feature vectors of 24 dimensions are extracted from the overall collected user data. For each sensor signal A_i, where i = {1, 2, …, 6} indicates the pressure sensor number, we transform signal A_i into the frequency domain by fast-Fourier transformation (FFT). In this way, we obtain for each sensor signal, two more time series A_{ij} with j ={t, f}, where t and f represent the time domain signal and the frequency domain signal, respectively. The feature vectors are extracted from a sliding window of size 30 with 50% overlap, which corresponds to one second of sensor time. For each window, the mean and variance are extracted from time series A_{ij} in both the time domain and the frequency domain. In other words, four-dimensional feature vectors are extracted for each sensor. Since in this experiment, we use six pressure sensors, the total feature dimensionality for each subject is 6 × 4, which results in 24-dimensional feature vectors. Thus, the training data consist of feature vectors of 24 dimensions, which is the columnwise combination of all the selected features from each sensor.

2.3 Activity recognition

Machine learning algorithms are utilized in this study to recognize the activities. There exist different machine learning and deep learning models in this field. However, this study is aimed at developing a low-cost, reliable smart chair that can recognize and classify five activities using the 24 selected features from the raw sensor readings. Therefore, three classification algorithms are employed to achieve this goal. These include random forest (RF), extremely randomized tree (ERT), and support vector machine (SVM).

RF assembles a number of decision tree classifiers and averages their predictions to improve the accuracy and control overfitting.(14,15) Each tree in the ensemble is built from a subsample drawn with replacement (i.e., a bootstrap sample) from the original dataset or training set that contains a collection of features.(16) In a classifier concept, the random vector $\boldsymbol{X} = (X_1, \ldots, X_p)^{\mathrm{T}}$ represents the real-valued input or features and the random feature Y represents the real-valued response, assuming an unknown joint distribution $P(X, Y)$. The aim is to find a prediction function $f(X)$ for predicting Y.(17)

In the ERT algorithm, randomness goes one step further in the way splits are computed. As in RFs, a random subset of candidate features is used, but instead of looking for the most discriminative thresholds, thresholds are drawn at random for each candidate feature and the best of these randomly generated thresholds is selected as the splitting rule. This usually allows a slightly greater reduction in the variance of the model, at the expense of a slightly greater increase in bias. The extra-trees algorithm builds an ensemble of the unpruned decision or regression trees following the classical top-down procedure. It has two main differences with other tree-based ensemble methods. It splits nodes by choosing cut-points fully at random and it uses the whole learning sample (rather than a bootstrap replica) to grow the trees.(18)

If the training dataset is also used as a test dataset during the testing of a classification function, the model will fail to classify anything useful on yet-unseen data resulting in overfitting. A common solution to it is to hold out part of the available data as a test set. In Scikit-learn, a random split into training and test sets can be quickly computed with the train_test_split helper function.(15) However, by doing this, valuable information that the learning algorithm could benefit from is withheld.

On the other hand, a smaller test set generates a more inaccurate estimation of the generalization error. Then, splitting a dataset into training and test sets is all about balancing this trade-off. The most commonly used splits are 60:40, 70:30, and 80:20, depending on the size of the initial dataset.(19)

Another way to split the dataset into training and testing sets is to use a cross-validation method called Leave One Out (LOO), in which one sample is used for testing and the remaining samples are used to train the model. The testing sample changes per iteration until all available samples in the dataset have been tested. However, this approach is time-consuming. In this study, 80% of the data for each user are used for training and 20% for testing.

The extracted features from each of the sensor data are processed with a RF classifier for classification. This classifier uses an ensemble learning method for classification. It forms an ensemble by building multiple decision trees during training and outputting the class that is the mode of the classes. This algorithm has many hyperparameters that have an effect on the model performance depending on how they are tuned. The parameters used in this research are listed in Table 1.

The algorithm of growing ERTs is similar to the RF, but there are still some differences. ERTs do not apply the bagging procedure to construct a set of training samples for each tree. The same input training set is used to train all the trees. Unlike RF, which finds the best among a random subset of variables, ERTs pick a node and split in the extreme case. The ERT

Table 1
Parameters of RF.

Parameter	Value	Meaning
Bootstrap	True	Bootstrap samples are used when building trees
Max features	Square root	The number of features to consider when looking for the best split
n estimators	20	The number of trees in the forest
Max depth	None	Nodes are expanded until all leaves are pure or until all leaves contain less than min_samples_split samples.

algorithm builds totally randomized trees whose structures are independent of the output values of the learning sample. The parameters used in this classifier are depicted in Table 2.

3. Experimental Results

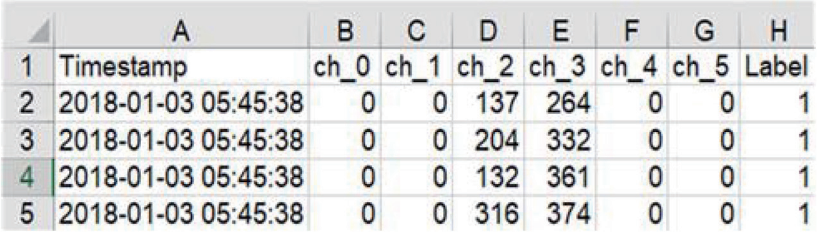
Eight different subjects participated in the data collection phase of this research. Each activity was performed by a subject for a duration of 10 min except for coughing. The activities were performed in the following sequence without taking a break: eating, working at the desk, napping, and watching TV. The subjects immediately switched to the next activity after 10 min. Considering that coughing is not a regular activity and it is spontaneous, it was performed separately for a 20 min duration. The subject does not cough continuously during this period; rather, they coughed freely at any time within the 20 min period. A video recording is taken of the participants performing the activities for reference. This makes it easy to trace the sensor data that correspond to the coughing activity instead of using all the redundant data. The resulting data from coughing are combined with the data from other activities for analysis.

The collected data are simultaneously stored along with the timestamp for ease of assigning labels to the data. These data are stored in CSV files and transmitted to the server to be processed separately. Each data file consists of six pressure sensor values, as depicted in Fig. 3. The column names ch_0 to ch_5 represent signal values from the six pressure sensors, where each column name corresponds to a pressure sensor. The label column is manually assigned with reference to the timestamp and the video stream of the activity. The pressure sensors are very sensitive and their values depend on the pressure applied. The pressures from 0 to 1024.

Figure 4 illustrates the proposed smart chair built from simple and sophisticated devices. Eight different subjects participated in this undertaking, performing the aforementioned activities. Each subject is seated on the chair in front of a desk with a computer and asked to perform the activities as they would do under normal circumstances.

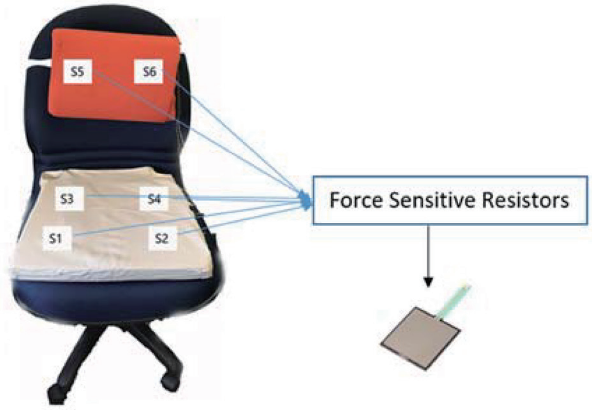
Table 2
Parameters of ERT.

Parameter	Value	Meaning
n estimators	20	The number of trees in the forest
Max depth	None	Nodes are expanded until all leaves are pure or until all leaves contain less than min_samples_split samples.



	A	B	C	D	E	F	G	H
1	Timestamp	ch_0	ch_1	ch_2	ch_3	ch_4	ch_5	Label
2	2018-01-03 05:45:38	0	0	137	264	0	0	1
3	2018-01-03 05:45:38	0	0	204	332	0	0	1
4	2018-01-03 05:45:38	0	0	132	361	0	0	1
5	2018-01-03 05:45:38	0	0	316	374	0	0	1

Fig. 3. Pressure sensor data.



"Force-sensitive resistors"

Fig. 4. (Color online) Prototype of the smart chair.

In order to effectively assess the performance of our smart chair, two phases of experiments were conducted, the user-dependent phase and the user-independent phase. In the user-dependent phase, 80% of each subject's data are included in the training dataset, and the remaining 20% are used for testing models. On the other hand, the user data for training and testing are different for the user-independent phase, meaning that the user data involved in training are not included in the testing data and vice versa.

3.1 User-dependent phase

The features extracted for all the participants consist of 45687 feature vectors. On applying the 80:20 split of datasets as mentioned above, the training and testing datasets consist of 34265 and 11422 feature vectors, respectively. The number of extracted features for each activity in the user-dependent phrase are outlined in Tables 3 and 4. The testing accuracy scores of RF, ERT, and SVM classifiers on each of the user data are depicted in Fig. 5. Both of RF and ERT attained accuracy scores above 95% on the individual activities. In terms of the classification accuracy scores, ERT outperformed the other models in this survey. Furthermore, the testing accuracy scores are diverse among the subjects because different subjects performed these activities in distinct styles and with various body orientations. The confusion matrix of ERT is illustrated in Table 5.

As portrayed in Table 5, we note that some activities were misclassified owing to contextual similarity. For example, working at the desk was misclassified as eating, napping, or watching, while napping was misclassified as eating or working at the desk.

3.2 User-independent phase

For the user-independent phase, the feature sizes for the activities included in training and testing are shown in Table 6. The total extracted features for the training and testing datasets consist of 8405 and 2909 feature vectors, respectively.

Table 3
Number of extracted features for each subject (training data).

Activity	Sub. 1	Sub. 2	Sub. 3	Sub. 4	Sub. 5	Sub. 6	Sub. 7	Sub. 8
Eating	1093	1052	1032	1114	1092	1089	1011	1121
Working	1023	1090	1109	960	1004	1046	1064	1007
Napping	946	1072	1110	1097	1069	1042	1106	973
Coughing	42	49	68	42	44	63	70	60
Watching	413	1054	1013	1426	1150	1041	1226	1169

Sub. means subject.

Table 4
Number of extracted features for each subject (testing data).

Activity	Sub. 1	Sub. 2	Sub. 3	Sub. 4	Sub. 5	Sub. 6	Sub. 7	Sub. 8
Eating	385	381	374	376	404	398	354	405
Working	327	400	388	348	354	348	388	359
Napping	307	306	329	335	298	295	339	298
Coughing	19	16	19	20	12	23	16	16
Watching	135	337	335	468	385	364	396	366

Sub. means subject.

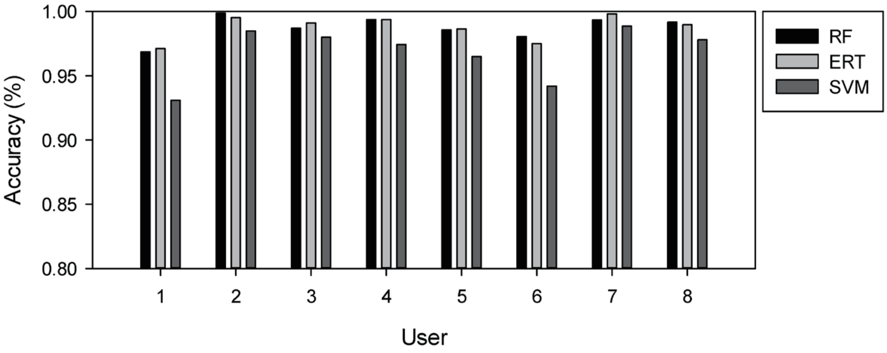


Fig. 5. Accuracy of models for each user.

Table 5
Confusion matrix of ERT.

	Eating	Working	Napping	Coughing	Watching
Eating	**0.9774**	0.0226	0.0000	0.0000	0.0000
Working	0.0429	**0.9539**	0.0032	0.0000	0.0004
Napping	0.0003	0.0049	**0.9921**	0.0000	0.0000
Coughing	0.0000	0.0080	0.0000	**0.9680**	0.0240
Watching	0.0003	0.0000	0.0000	0.0004	**0.9993**

The rows represent the ground truth of the five activities and the columns represent the predictions.

Table 6
Feature sizes for all the activities included in training and testing data.

Activity	Training	Testing
Eating	2154	742
Working	2080	679
Napping	2085	685
Coughing	98	32
Watching	1988	771

Table 7
Confusion matrix of ERT.

	Eating	Working	Napping	Coughing	Watching
Eating	**0.9559**	0.0427	0.0000	0.0000	0.0014
Working	0.0316	**0.9638**	0.0045	0.0000	0.0004
Napping	0.0043	0.0028	**0.9929**	0.0000	0.0000
Coughing	0.0000	0.0000	0.0000	**0.1000**	0.0000
Watching	0.0000	0.000	0.0000	0.0000	**0.1000**

The rows represent the ground truth of the five activities and the columns represent the predictions.

The classification performances of the three models, RF, ERT, and SVM, in the user-independent phase are 0.978, 0.979, and 0.962, respectively. ERT also achieved a higher classification accuracy than RF and SVM; its corresponding confusion matrix is depicted in Table 7. This is quite reasonable considering that the data from each participant are included in both training and testing on the 80:20 split basis. Nonetheless, the results attained in the user-independent phase are promising as they show a better generalization attribute since the user data included in training are separate from those contained in testing.

4. Conclusions

We developed a smart chair that can detect and classify five common activities performed in a sedentary position. The system was composed of six pressure sensors, Raspberry Pi, and an office chair. Four pressure sensors were mounted on the seat cushion to detect information of the participants while seated and two on the backrest to capture the details when the user leans back. The data collected via Raspberry Pi were sent to the server for preprocessing and analysis.

Two phases of experiments were conducted on the collected data upon submission to the server, namely, user-dependent and user-independent experiments. In the two phases of experiments, for each sensor reading of data from each participant, the mean and the variance are extracted from a window size of 30 with 50% overlap. These features are extracted from both the frequency and time domains, resulting in four features per sensor. Since we employed six sensors, 24-dimensionality feature vectors are extracted for each user.

The RF and ERT classifiers demonstrated a very high classification performance during the experiments, the highest being attained by ERT. ERT obtained up to 98% in the user-dependent phase and 97% in user-independent phase. The results obtained by the classifiers on the five activities showed that the proposed algorithm outperformed all the others mentioned in the literature.

Acknowledgments

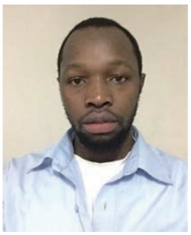
This work was supported by the Ministry of Science and Technology (Grant number: MOST 105-2622-E-155-015-CC3).

References

1 T. Kasteren, G. Englebienne, and B. Kröse: Pers. Ubiquitous Comput. **14** (2010) 489.
2 T. S. Barger, D. E. Brown, and M. Alwan: IEEE Trans. Syst. Man Cybern. Part A Syst. Humans **35** (2005) 22.
3 M. Nambu, K. Nakajima, A. Kawarada, and T. Tamura: Proc. 2000 IEEE Conf. EMBS (2000) 79.
4 N. Noury, A. Dittmar, C. Corroy, R. Baghai, J. Weber, D. Blanc, F. Klefstat, A. Blinovska, S. Vaysse, and B. Comet: Proc. 2004. IEEE Conf. EMBS **2** (2004) 3266.
5 C.-C. Yang and Y.-L. Hsu: Sensors **10** (2010) 7772.
6 D. W. Kang, J. S. Choi, J. W. Lee, S. C. Chung, S. J. Park, and G. R. Tack: Disability and Rehabil.: Assistive Technol. **5** (2010) 247.
7 Y. Charlon, W. Bourennane, F. Bettahar, and E. Campo: IRBM **34** (2013) 60.
8 N. Zouba, F. Bremond, and M. Thonnat: Proc. 2010 IEEE Conf. AVSS (2010) 278.
9 N. Zouba, F. Bremond, and M. Thonnat: Proc. 2009 IEEE Conf. AVSS (2009) 98.
10 H. Z. Tan, L. A. Slivovsky, and A. Pentland: IEEE/ASME Trans. Mechatron. **6** (2001) 261.
11 T. Fu and A. Macleod: Proc. 2014 Int. Conf. Intell. Environments (2014) 211.
12 R. Kumar, A. Bayliff, D. De, A. Evans, S. K. Das, and M. Makos: Proc. 2016 IEEE Conf. SMARTCOMP (2016) 1.
13 A. Sadun, J. Jalani, and J. Sukor: Proc. 2016 First Int. Workshop on Pattern Recognit. **10011** (2016) 1001112.
14 L. Breiman: Mach. Learn. **45** (2001) 5.
15 F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, and V. Dubourg: J. Mach. Learn. Res. **12** (2011) 2825.
16 P. Geurts, A. Irrthum, and L. Wehenkel: Mol. Biosyst. **5** (2009) 1593.
17 A. Cutler, D. R. Cutler, and J. R. Stevens: Random Forests: Ensemble Machine Learning (Springer, Heidelberg, 2012) p. 157.
18 P. Geurts, D. Ernst, and L. Wehenkel: Mach. Learn. **63** (2006) 3.
19 S. Raschka: Python Machine Learning (Packt Publishing Ltd., Birmingham, 2015).

About the Authors

Chien-Cheng Lee received his Ph.D. degree in electrical engineering from National Cheng Kung University, Tainan, Taiwan, in 2003. Dr. Lee is currently an assistant professor in the Department of Electrical Engineering, Yuan Ze University, Taoyuan, Taiwan. He was a research visitor at Telcordia Inc. (formerly Bellcore), NJ, USA, from Oct. 2007 to Jan. 2008. He is one of the guest editors for a special issue on Signal Processing for Applications in Healthcare Systems for EURASIP Journal on Advances in Signal Processing, 2008. His research interests include image processing, pattern recognition, and machine learning.



Lamin Saidy received his B.S. degree from the University of The Gambia, Gambia, in 2014 and his M.S. degree from Yuan Ze University, Taiwan, in 2018. From 2015 to 2016, he was a teaching assistant at University of The Gambia, Gambia. Since 2016, he has been a master's student at Yuan Ze University. His research interests are in machine learning, deep learning, and IoT.

Fitri received her B.S. degree from Polytechnic of Caltex Riau, Indonesia, in 2015 and her M.S. degree from Yuan Ze University, Taiwan, in 2018. Since 2016, she has been a master student at Yuan Ze University. Her research interests are in machine learning, IoT, and data analysis.