

# Classification methods performance on human activity recognition

I A Bustoni<sup>1</sup>, I Hidayatulloh<sup>2</sup>, A M Ningtyas<sup>3</sup>, A Purwaningsih<sup>1</sup> and SN Azhari<sup>1</sup>

<sup>1</sup> Departement of Computer Sciences and Electronics, Universitas Gadjah Mada, Indonesia

<sup>2</sup> Department of Electronics and Informatics Education, Faculty of Engineering, Universitas Negeri Yogyakarta, Indonesia

<sup>3</sup> Department of Health Information and Services, Vocational College, Universitas Gadjah Mada, Indonesia

E-mail: isna@ugm.ac.id

**Abstract.** There are around 650 million people from all over the world who lived with disabilities. One of the fundamental rights of people with disabilities is the existence of a companion to supervise his activity. Meanwhile, the use of mobile phones for monitoring the activities of people with disabilities has been widely carried out. The human activity monitoring mobile application requires human activity recognition methods that provide high accuracy, precision, and recall to reduce the error rate of the activity estimation. Some researches use machine learning algorithms like K-Nearest Neighbour (k-NN) algorithm, Artificial Neural Networks (ANN), Support Vector Machine (SVM), and Random Forest for human activity recognition methods. However, the results of these studies have not been compared apples to apples. Therefore, this study presents a performance comparison of SVM, KNN, and Random Forest machine learning methods. Based on our findings, the SVM method with Support Vector Classifier (SVC) and Radial Basis Function (RBF) kernels can achieve the highest precision and recall, 87% and 85% respectively. The fastest processing time is obtained using the SVM method with the Stochastic Gradient Descent. However, in general, the best performance is shown by Random Forest. The Random Forest method with a depth of 100 and 300 trees can reach an accuracy of 96% within 0.45 minutes.

## 1. Introduction

The human right for people with disabilities in the aspect of economic, political, social, or cultural rights are still less than optimal. Based on the World Health Organization (WHO) report in 2011, around 650 million productive age world population endure disabilities[1]. Meanwhile, in Indonesia, it was around 6 million of the population[2] as shown on a survey report by Survei Sosial Ekonomi Nasional (Susenas) in 2012. Facilities to support the needs of persons with disabilities are still inadequate[3][4]. One of them is the need for a companion to supervise their activity. Protection and supervision of persons with



Content from this work may be used under the terms of the [Creative Commons Attribution 3.0 licence](https://creativecommons.org/licenses/by/3.0/). Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

disabilities need to be done 24 hours per day to protect them from injury, danger, or accident [5]. On the other hand, the utilization of mobile phones for monitoring their activity has been widely carried out [6], [7]. Human activity data can be recorded using Accelerometer and Gyroscope sensors in the smartphone.

Research on Human Activity Recognition (HAR) will produce information about a person's behaviour and actions based on sensor data [8]–[10]. Furthermore, a disabled assistance system as an activity monitoring application developed by utilizing this information. This human activity monitoring application requires human activity recognition methods that provide high accuracy, precision, and recall of the activity classification. Several previous studies already mention some machine learning methods to classify human activities[11]. The top well-known methods, including the K-Nearest Neighbour (k-NN) algorithm, Artificial Neural Networks (ANN), and Support Vector Machine (SVM) [10], [12], [13] are tested. On the other hand, the Random Forest methods is envisioned to provide high accuracy with more reliable speed performance for data mining, especially classification with sophisticated features[14], [15].

Unfortunately, the results of these studies have not been compared apple to apple in terms of performance including accuracy, precision, recall, F1-Score, and computational speed. Therefore, this study proposes a performance comparison of machine learning methods: SVM, K-NN, and Random Forest in classifying sensor data on human motion activities, which include, accuracy, precision, recall, and computational speed so that the most effective method is found. In this study, the SVM method will implement Stochastic Gradient Descent, and Support Vector Classifier with RBF kernel.

### *1.1. Literature review*

Human Activity Recognition was first proposed using the well-known machine learning classification and achieved an accuracy of up to 84% concerning the activities involved [16]. Other research has also been carried out by developing a sequential classification formed by Gaussian Continuous Emission HMM (Gaussian cHMM)[17]. In subsequent developments, the concept of Hardware-Friendly Support Vector Machine (HF-SVM) was introduced. This method utilizes a fixed-point in the feed-forward phase of SVM and extends this model to multiclass classification[18]. This research evolved later with the addition of binary-classifier with a one-vs-all approach [9].

Human Activity Recognition is also used to detect movement disorders in a smarter interactive cognitive environment [13]. Ortiz also developed the TAHAR (Transition-Aware Human Activity Recognition) framework to overcome the classification of activities outside the pre-made class and to cope with the postural transition [19], [20]. Generally, records of human activity obtained by applying the supervised learning algorithm, although semi-supervised and unsupervised methods was also proposed[21]. Some supervised learning methods with Frequentist and Bayesian models involve predictive models such as binary decision trees and threshold-based classifiers[22]. These models including a probabilistic classification methods such as Naive Bayes and Hidden Markov Models [23], k-Nearest Neighbours (k-NN), Artificial Neural Networks (ANN) and Support Vector Machine (SVM)[10].

The modified random forest algorithm was implemented using importance score driven random forest to classify Human Activities and Postural Transitions (HAPT)[15],[24]. Further development of this method was adding max-min features and key poses [25]. Unlike the previous methods, the Logistic Regression method uses a combination of Statistics and Supervised Learning methods. This method provides a reasonably high accuracy because it has a characteristic point of probability (selected results with the highest probability) compared to the ANN method [26][14].

### 1.2. Data acquisition

Data was obtained from the UCL Machine Learning repository site. This research uses human movement sensor dataset[27]. This dataset was built from a recording of 30 subjects that perform three basic activities (standing, sitting, and lying down) and dynamic activities (walking, walking down, walking up). This experiment included postural transitions that occurred between static postures, namely stand-to-sit, sit-to-stand, sit-to-lie, lie-to-sit, stand-to-lie, and lie-to-stand. All subjects carried a smartphone (Samsung Galaxy S II). The smartphone is equipped with a sensor at the waist. Furthermore, 3-axial linear acceleration and 3-axial angular velocity were collected at a constant speed of 50 Hz by using an accelerometer and gyroscope. In this study, the data are classified into 12 classes: Walking, Walking\_Upstairs, Walking\_Downstairs, Sitting, Standing, Laying, Stand\_to\_Sit, Sit\_to\_Stand, Sit\_to\_Lie, Lie\_to\_Sit, Stand\_to\_Lie, and Lie\_to\_Stand.

## 2. Methods

### 2.1. *k*-Nearest Neighbours (k-NN)

The *k*-Nearest Neighbours (k-NN) classifier is one of the well-known and simplest classifiers. The main idea of this classifier is to classified based on the similarity to the known, trained, labeled samples by computing the distance between the unknown sample and all labeled samples [28], [29]. The *k*-nearest sample is selected as the basis of the classification. The unknown sample will be assigned to the class with most samples among the *k*-nearest samples. This k-NN classifier depends on; first, the numbers of the neighbors (*k* nearest), changing this parameter will change the classification results. Second, k-NN is a supervised classifier; it means that the K-NN classifier needs a set of training samples to train the model[29]. Adding and removing the training samples will be affected by the model and the final decision of the k-NN classifier. Last, a distance matrix. Euclidean distance is often used to measure the distance between two samples. The formula of Euclidean distance denoted in Eq.1.

$$d_{(xi,xj)} = \sum_{k=1}^d (x_{ik} - x_{jk})^2 \quad (1)$$

where  $d_{(xi,xj)}$  represents the distance between two samples  $xi$  and  $xj$ .  $(xi, xj) \in \mathcal{R}^m$ ,  $xi = \{xi1, xi2, \dots, xim\}$ ,  $m$  is the number of attributes of samples.

### 2.2. Support Vector Machine (SVM)

SVM is a supervised classifier. It is similar to the k-NN classifier because they need a training set to build a model for classification. SVM is a good classifier for data analysis and classification, because it has a fast learning speed, even in large dataset. SVM usually used for the binary classifier, and it is used to classify the data with a similar feature value. Irrelevant and relevant vectors are separated by using hyperplane. Hyperplane also works to separate the data in the feature space. It can reduce the problem of overfitting the training data. The hyperplane also is known as the decision boundary[30]–[32].

The classes are assumed to identified as  $\pm 1$  and the hyperplane is estimate as  $y=0$ , using the equation(2) below:

$$y = \sum_{i=1}^N w_i x_i + b = x_i w + b \quad (2)$$

Where  $x_i$  is the input, and  $w$  is the weight vector, while  $b$  is the offset. In this research, we use Support Vector Classifier with RBF kernel, and SVM with Stochastic Gradient Descent.

### 2.3. Random Forest

Random forest is one of a machine learning method that is composed of decision trees. Random forest combines many decision trees to classifying and predicting. Each decision tree is built up by row sampling and column sampling [33][34], [35]. The random forest decision trees are obtained from bootstrap. Bootstrap is a model integration method with a replacement method. When building a model for the classification, bootstrap does a random sampling to extracts a fixed number of samples from the training set and put it back into the training set after the sampling.

For example in a random sample with sample  $q$ . The probability of samples being collected and not collected each time is  $1/q$  and  $1-1/q$  respectively. With  $L$  random sampling times carried out, the probability of the sample not being collected is  $(1-1/q)^L$  where  $L$  converges to  $\infty$ ,  $(1-1/q)^L$  converges to  $1/e = 0.368$ . Thus, there will be duplicate samples. Also, there will be  $1/3$  individuals who will be left behind in the new sample. Data that has not been extracted is called an "Out-of-Bag" individual. Out-of-bag (OOB) data errors are also called OOB errors. The following is the formula:

$$OE = XNB \quad (3)$$

In Equation 1,  $X$  shows the number of errors in the test results for NB Amount of Data.  $NB$  shows the number of OOBs and also the categories known from each data. Utilizing the Gini Index, which describes the level of impurity of the model, by the CART algorithm, a decision tree is generated. A smaller Gini index means lower impurity. For classification problems, the probability of the  $K$ -th category is  $p_k$  for the  $K$  category, the Gini index formula is represented as in Equation(4).

$$\text{Gini}(D) = \sum_k = 1Kp_k(1 - p_k) = 1 - \sum_k = 1Kp_k^2 \quad (4)$$

The Gini index is used as the basis for selecting features from the decision tree. The formula is shown in Equation (5).

$$\Delta\text{Gini}(A) = \text{Gini}(D) - \text{GiniA}(D) \quad (5)$$

The Gini index getting the maximum value will be selected for split characteristics, where the node is used as a split condition.

### 3. Performance assessment

This study conducted a performance assessment using precision, recall, accuracy and F1-Score [36]. In order to measure computing speed, the entire training and testing process of all methods is carried out on the same computer. We use computing devices with the following specifications: Intel Core i5-6200u (2.8 GHz), NVIDIA GEFORCE 930MX, 4GB RAM. Furthermore, the computational process is carried out using Python version 3.7.3.

### 4. Implementation & result

The data was divided into two parts with a composition of 70% training data and 30% testing data. In other words, there are 7500 data for training data and 3,215 data for testing. This dataset also has 561 features which will be used to classify data into 12 classes.

This study uses the SVM Multiclass with the "one-against-one" approach. This research will also compare the performance of Support Vector Classifier and Stochastic Gradient Descent. Furthermore, for the implementation of the  $k$ -Nearest Neighbors algorithm, a  $k$  value is searched using Euclidean Distance by Eq.1. In order to search for  $k$  values, we calculated the distance between each point of the

test data with the training data. We compare the performance of  $k = 12$  and  $k = 20$ . In addition to using SVM and KNN, we also use the Random Forest method. The implementation of the Random Forest method will use the parameters of 300 trees with a depth of 100 and 100 trees with a depth of 300.

The performance comparison and test results are shown in tables 1 and 2 as follows:

**Table 1.** Precision and recall comparison

	Model	1	2	3	4	5	6	7	8	9	10	11	12	Avg.
Precision	SVM(SGD)	0.94	0.99	0.96	0.95	0.92	1.00	0.81	1.00	0.68	1.00	0.49	0.49	0.85
	SVM(RBF)	0.96	0.98	0.99	0.97	0.91	0.99	0.95	0.91	0.61	0.74	0.67	0.73	0.87
	KNN (12)	0.85	0.88	0.97	0.88	0.84	1.00	0.84	1.00	0.62	0.64	0.79	0.80	0.84
	KNN(20)	0.85	0.87	0.97	0.91	0.85	1.00	0.83	1.00	0.60	0.63	0.71	0.72	0.83
	RF(100,300)	0.98	0.97	0.97	0.96	0.96	1.00	0.76	0.88	0.82	0.47	0.74	0.60	0.84
	RF(300,100)	0.99	0.97	0.97	0.95	0.95	1.00	0.81	1.00	0.86	0.50	0.73	0.64	0.86
Recall	SVM(SGD)	0.99	0.88	0.99	0.90	0.96	1.00	0.74	0.90	0.60	0.04	0.73	0.86	0.80
	SVM(RBF)	0.99	0.97	0.97	0.89	0.98	1.00	0.78	1.00	0.63	0.68	0.67	0.68	0.85
	KNN(12)	0.98	0.92	0.78	0.81	0.90	0.99	0.70	0.90	0.89	0.84	0.45	0.43	0.80
	KNN(20)	0.99	0.92	0.76	0.82	0.93	0.99	0.65	0.80	0.86	0.76	0.41	0.46	0.78
	RF(100,300)	0.98	0.98	0.96	0.95	0.97	1.00	0.80	0.70	0.67	0.42	0.83	0.56	0.82
	RF(300,100)	0.98	0.98	0.97	0.94	0.96	1.00	0.85	0.70	0.67	0.53	0.86	0.52	0.83

**Table 2.** General performance assesment

Model	Avg Precision	Avg Recall	Accuracy	F1-Score	Time Proc. (min)
SVM(SGD)	0.85	0.80	0.94	0.94	0.06
SVM(SVC-RBF)	0.87	0.85	0.95	0.95	7.90
KNN (12)	0.84	0.80	0.89	0.89	0.85
KNN(20)	0.83	0.78	0.89	0.89	0.83
RF(100,300)	0.84	0.82	0.96	0.96	1.40
RF(300,100)	0.86	0.83	0.96	0.96	0.45

The SVM method with the Linear SVC approach (Support Vector Classifier) has the best precision and recall performance, which is 87% and 85% as shown in Table 2. The best implementation of the SVM method is obtained by using the RBF kernel with the gamma of 0.0001 and the value of  $C = 1000$ . This indicates that the data has not been entirely linearly separated. However, the best accuracy and F1-Score is shown by the Random Forest method with an accuracy value and an F1-Score of 96%. Furthermore, the method with the shortest processing time is SVM with Stochastic Gradient Descent.

## 5. Discussion

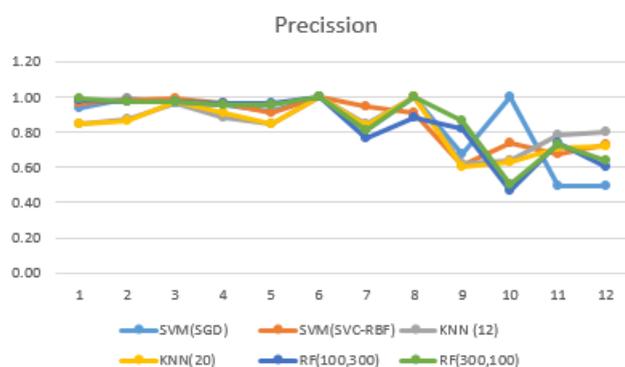


Figure 1. Precision Comparison

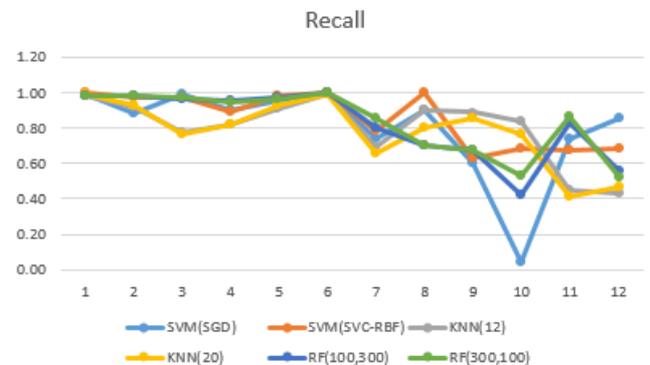


Figure 2. Recall Comparison

The SVM method with the Stochastic Gradient Descent has the shortest computational time but also produces the smallest precision, recall, accuracy, and F1-Score. Meanwhile, the SVM method with SVC and RBF kernels has the longest computational time (7.9 minutes). However, this is commensurate to its performance. As a comparison, Random Forest with both 300 trees and 100 trees produces a high accuracy performance and an F1-Score of 96%. This performance is obtained within 1.4 minutes and 0.45 minutes.

Based on figure 1 and 2, performance of all method decreased while classifying dataset into class 7 to 12. It is reasonable since class 7-12 are transitional position class. In addition, the worst average precision and recall is belong to the class 12 which is Lie-to-Stand class. The next challenge will be how to increased accuracy, precision and recall of transitional class classification. Transitional Position class could contain a fuzzy point because the data point likely to be not linearly separated. When data point quite near to a static class, it is extremely possible for classification method to assign this data point into nearest static class. In future research, this "cryptic" data challenge can be overcome by adding fuzzy methods. The fuzzy method implementation for data classification was also carried out by the study [37]. This study adds a fuzzy approach to Seasonal Autoregressive Integrated Moving Average model [38]. Moreover, our study also used all features contained in the dataset (561 features). Future studies can also consider feature selection and feature scaling to optimize the classification process.

## 6. Conclusion

The Random Forest method can produce the best performance, which is an F1-Score of 96% with a faster processing speed than the SVM method with SVC and RBF kernel. The Random Forest method also has a characteristic of being suitable for data with massive features. It also affirms the accuracy of this method in both using 100 or 300 trees. On the other hand, the Random Forest also shows a significant difference in processing time for the depths of 100 and 300 trees. Thus it can be said that the choice of depth affects the computational time but does not significantly affect accuracy.

## 7. References

- [1] W. Health Organization, "World report on disability 2011. World Health Organisation and The World Bank," vol. 91, p. 549, 2011.
- [2] Kementrian Kesehatan RI, "Situasi Penyandang Disabilitas," 2014.
- [3] Frieda Isyana Putri, "Ada 4 Juta Difabel di Indonesia Belum Rasakan Fasilitas yang 'Ramah.'"

- [Online]. Available: <https://health.detik.com/berita-detikhealth/d-3991194/ada-4-juta-difabel-di-indonesia-belum-rasakan-fasilitas-yang-ramah>. [Accessed: 06-Mar-2019].
- [4] I. A. Bustoni, I. Hidayatulloh, and A. Azhari, "Multidimensional Earcon Interaction Design for The Blind: a Proposal and Evaluation," in *2018 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*, 2018, no. November, pp. 21–22.
- [5] "In-Home Supportive Services Protective Supervision," 2017. .
- [6] J. Usharani, "Human Activity Recognition using Android Smartphone," in *1st International Conference on Innovations in Computing & Networking*, 2016, pp. 191–197.
- [7] A. Bayat, M. Pomplun, and D. A. Tran, "A study on human activity recognition using accelerometer data from smartphones," *Procedia Comput. Sci.*, vol. 34, no. C, pp. 450–457, 2014.
- [8] B. P. P. Clarkson, "Life Patterns: structure from wearable sensors," *Dspacemitedu*, pp. 2003–2003, 2002.
- [9] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "A Public Domain Dataset for Human Activity Recognition Using Smartphones," *Can J Cardiol*, vol. 20, no. 5, pp. 555–556, 2004.
- [10] A. M. Khan, Y. K. Lee, S. Y. Lee, and T. S. Kim, "Human activity recognition via an accelerometer-enabled-smartphone using Kernel Discriminant Analysis," *2010 5th Int. Conf. Futur. Inf. Technol. Futur. 2010 - Proc.*, 2010.
- [11] A. Purwaningsih, "Comparison Of Support Vector Machine And K-Nearest Neighbors Methods For Classification In Human Motion Sensor Data," Universitas Gadjah Mada, 2013.
- [12] S. Ougiaroglou, A. Nanopoulos, A. N. Papadopoulos, Y. Manolopoulos, and T. Welzer-Druzovec, "Adaptive k-Nearest-Neighbor Classification Using a Dynamic Number of Nearest Neighbors," in *11th East European Conference, ADBIS 2007 Varna, Bulgaria, September 29-October 3, 2007*, 2007, pp. 66–82.
- [13] J. L. Reyes-Ortiz, A. Ghio, D. Anguita, X. Parra, J. Cabestany, and A. Catal, "Human activity and motion disorder recognition: Towards smarter interactive cognitive environments," *Eur. Symp. Artif. Neural Networks, Comput. Intell. Mach. Learn. ESANN*, no. April, pp. 24–26, 2013.
- [14] R. Madarshahian and J. M. Caicedo, "Human Activity Recognition Using Multinomial Logistic Regression," in *Model Validation and Uncertainty Quantification*, 2015, vol. 3, no. 2015, pp. 363–372.
- [15] M. T. Uddin, M. M. Billah, and M. F. Hossain, "Random forests based recognition of human activities and postural transitions on smartphone," *2016 5th Int. Conf. Informatics, Electron. Vision, ICIEV 2016*, no. 978, pp. 250–255, 2016.
- [16] L. Bao and S. S. Intille, "SKMBT\_36316102014450.pdf," pp. 1–17, 2004.
- [17] A. Mannini and A. M. Sabatini, "Machine learning methods for classifying human physical activity from on-body accelerometers," *Sensors*, vol. 10, no. 2, pp. 1154–1175, 2010.
- [18] D. Anguita, A. Ghio, S. Member, S. Pischiutta, and S. Ridella, "A Hardware-friendly Support," no. 3, pp. 1–5, 2007.
- [19] J. L. Reyes-Ortiz, L. Oneto, A. Samà, X. Parra, and D. Anguita, "Transition-Aware Human Activity Recognition Using Smartphones," *Neurocomputing*, vol. 171, pp. 754–767, 2016.
- [20] J. L. Reyes-Ortiz, L. Oneto, A. Ghio, A. Samà, D. Anguita, and X. Parra, "Human activity recognition on smartphones with awareness of basic activities and postural transitions," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 8681 LNCS, pp. 177–184, 2014.
- [21] M. Stikic, K. Van Laerhoven, and B. Schiele, "Exploring Semi-Supervised and Active Learning for Activity Recognition," pp. 81–88, 2008.
- [22] M. Ermes, J. Parkka, and L. Cluitmans, "Advancing from offline to online activity recognition with wearable sensors," pp. 4451–4454, 2009.
- [23] E. M. Tapia *et al.*, "Real-time recognition of physical activities and their intensities using wireless accelerometers and a heart rate monitor," *Proc. - Int. Symp. Wearable Comput. ISWC*, pp. 37–

- 40, 2007.
- [24] Z. Feng, L. Mo, and M. Li, "A Random Forest-based ensemble method for activity recognition," *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. EMBS*, vol. 2015-Novem, pp. 5074–5077, 2015.
- [25] U. M. Nunes, D. R. Faria, and P. Peixoto, "A human activity recognition framework using max-min features and key poses with differential evolution random forests classifier," *Pattern Recognit. Lett.*, vol. 99, pp. 21–31, 2017.
- [26] H. Vellampalli, "Physical Human Activity Recognition Using Machine Learning Algorithms Physical Human Activity Recognition Using Machine Learning Algorithms A dissertation submitted in partial fulfillment of the requirements of," 2017.
- [27] J. L. Reyes-Ortiz, L. Oneto, A. Samà, X. Parra, and D. Anguita, "Transition-Aware Human Activity Recognition Using Smartphones," *Neurocomputing*, vol. 171, pp. 754–767, 2016.
- [28] A. Tharwat, H. Mahdi, M. Elhoseny, and A. E. Hassanien, "Recognizing human activity in mobile crowdsensing environment using optimized k-NN algorithm," *Expert Syst. with Apl.*, vol. 107, pp. 32–44, 2018.
- [29] J. X. Yu and R. Goebel, *Advanced Data Mining and Applications*. Guilin, China: Spr, 2014.
- [30] G. Karthick and R. Harikumar, "Comparative Performance Analysis of Naive Bayes and SVM classifier for Oral X-ray images," in *2017 4th International Conference on Electronics and Communication Systems (ICECS)*, 2017, vol. 17, pp. 88–92.
- [31] D. Saraswathi and E. Srinivasan, "Performance Analysis of Mammogram," in *International Conference on Inventive Systems and Control (ICISC-2017)*, 2017, pp. 1–5.
- [32] Parveen and A. Singh, "Detection of Brain Tumor in MRI Images , using Combination of Fuzzy C-Means and SVM," in *2015 2nd International Conference on Signal Processing and Integrated Networks (SPIN)*, 2015, pp. 98–102.
- [33] H. Wang and Y. Liu, "Random Forest and Bayesian Prediction for Hepatitis B Virus Reactivation," *2017 13th Int. Conf. Nat. Comput. Fuzzy Syst. Knowl. Discov.*, no. 81402538, pp. 2060–2064, 2017.
- [34] U. Aprilliani and Z. Rustam, "Osteoarthritis Disease Prediction Based on Random Forest," *2018 Int. Conf. Adv. Comput. Sci. Inf. Syst.*, pp. 237–240, 2018.
- [35] X. Ye, X. Wu, and Y. Guo, "Real-time Quality Prediction of Casting Billet Based on Random Forest Algorithm," *2018 IEEE Int. Conf. Prog. Informatics Comput.*, pp. 140–143, 2018.
- [36] C. Goutte and E. Gaussier, "A Probabilistic Interpretation of Precision, Recall and F-Score, with Implication for Evaluation," *Lect. Notes Comput. Sci.*, vol. 3408, pp. 345–359, 2005.
- [37] I. Alfi, I. Hidayah, A. Erna, and I. Hidayatulloh, "Fuzzy Logic Tsukamoto for SARIMA On Automation of Bandwidth Allocation," *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 11, pp. 392–397, 2017.
- [38] A. E. Permanasari, I. Hidayah, and I. A. Bustoni, "Forecasting Model for Hotspot Bandwidth Management at Department of Electrical Engineering and Information Technology UGM," *Int. J. Appl. Math.*, vol. 53, no. 4, 2015.