

Unknown On-body Device Position Detection Based on Ensemble Novelty Detection

Mitsuaki Saito* and Kaori Fujinami

Department of Computer and Information Sciences, Tokyo University of Agriculture and Technology,
2-24-16 Naka-cho, Koganei, Tokyo 184-8588, Japan

(Received September 2, 2019; accepted October 29, 2019)

Keywords: novelty detection, smartphone, on-body device localization, ensemble learning

In recent years, on-body device position recognition has attracted a lot of attention from the ubiquitous computing community with a view to providing reliable services to users. The existing work has focused on the recognition of classes included in a training dataset, but handling a new position that the recognition system does not know is still impossible. The unknown position should be handled in an appropriate way to avoid incorrect behavior and adapt to each user's way of carrying the device. In this article, we propose a new detection method based on the ensemble learning principle, in which the final results are obtained from a collection of judgments by a weak novelty detector. We devise a method of finding a threshold that maximizes overall accuracy, rather than a mere majority vote. This method is evaluated with three datasets and various conditions to confirm the effectiveness of ensemble novelty detection and the threshold estimation method.

1. Introduction

People carry on-body devices in a variety of ways, including in pockets and bags.⁽¹⁾ These ways play an important role in the usability of an on-body device and the quality of sensor-dependent services that facilitate human–human communication, reduce unnecessary energy consumption, and automatically select an appropriate notification method.^(2–4) On-body device position recognition is gaining attention in ubiquitous computing communities that use multiclass classification techniques and where the number of positions is fixed before use.^(2,5–8) However, in reality, a variety of on-body device positions are available, and a user may carry the device in a position not originally intended for the system, i.e., an unknown position. In this case, the classifier performs incorrect recognition because the current position of the device is not in the range of recognition.

Therefore, the aim of this study is to detect unknown positions using novelty detection technology. When a position that is difficult to classify into any known position type is detected in the use stage, it is not classified into a known position type, but it is rather judged as

*Corresponding author: e-mail: perorokey@gmail.com
<https://doi.org/10.18494/SAM.2020.2585>

an unknown position so that appropriate processing can be invoked. In our previous work, we proposed a framework to detect and add unknown positions through the active involvement of the user,⁽⁹⁾ in which three types of single novelty detection methods were compared in terms of detection accuracy. In this article, we propose an ensemble novelty detection method based on the principle of ensemble learning, such as random forest,⁽¹⁰⁾ and apply it to unknown position detection. In the proposed method, novelty detection is performed with multiple weak detectors trained with different samples that accept different randomly selected features. Multiple results from weak detectors are integrated to obtain the final judgment on known or unknown positions. We propose a method of estimating an appropriate threshold for the judgment, rather than a mere majority vote. This estimation method is designed to be effective without any user involvement, rather than using a dataset to train weak detectors. The feasibility of the proposed method is evaluated for unknown position detection in on-body device position recognition under various conditions, such as different datasets and combinations of known positions.

The remainder of this article is organized as follows. In Sect. 2, the work related to on-body position recognition and examples of novelty detection applications are examined. In Sect. 3, the proposed method is described with a preliminary experiment to assess the effectiveness of ensemble novelty detection, and a new way to detect unknown positions more accurately is devised. In Sect. 4, the evaluation method is presented. In Sect. 5, the experimental results and the effectiveness of the proposed method are discussed. Finally, in Sect. 6, the article is concluded.

2. Related Work

Table 1 shows the storing positions of mobile devices found in the literature, which include trouser pockets, chest pockets, shoulder bags, handbags, and the hand. Here, the accuracy generally cannot be compared because experimental conditions and verification methods differ in each work. However, literature all focus on recognizing the position of the device in one of the predefined positions. Hence, the system misrecognizes data from an unknown position as one of the known positions. However, once the system knows that the data are obtained from

Table 1
Examples of on-body position recognition.

Literature	Positions supported in the work
Fujinami <i>et al.</i> ⁽⁶⁾	neck (hanging), chest pocket, jacket pocket, trouser front/back pockets, bag (backpack, handbag, and shoulder bag), hand (calling, watching the screen in the portrait direction, and swinging during walking)
Fujinami ⁽²⁾	neck (hanging), chest pocket, jacket pocket, trouser front/back pockets, bag (backpack, handbag, shoulder bag, and messenger bag)
Alanezi and Mishra ⁽⁷⁾	jacket pocket, trouser front/back pockets, desk, hand (calling, watching the screen in the portrait direction, and swinging during walking)
Shi <i>et al.</i> ⁽¹¹⁾	chest pocket, trouser front/back pockets, hand
Bieshaar ⁽¹²⁾	jacket pocket, trouser front/back pockets, backpack
Sztyler and Stuckenschmidt ⁽¹³⁾	head, chest, upper arm, waist, forearm, thigh, shin
Wiese <i>et al.</i> ⁽¹⁴⁾	pocket, bag, hand, away from human
Yang and Wang ⁽¹⁵⁾	jacket pocket, trouser pocket, bag, hand

an unknown position, it can take appropriate action, such as discard the result and ask the user to label the data for registering the detected position to recognition targets.

Novelty detection is a technique that determines if a test sample belongs to a known or trained class and consists of the following steps:

1. Novelty model construction provides a set of samples belonging to known classes used to train a model of known classes as an inverse representation of the novelty model.
2. Unknown sample detection projects a test sample to the model space and identifies it as a known or unknown sample.

Novelty detection is applied to various themes, such as images on the web, text mining, and network security.^(16–18) Additionally, novelty detection using the inertial sensor data of on-body devices has been conducted in human activity recognition. Yin *et al.* studied human activity recognition, and unknown activities were detected using novelty detection.⁽¹⁹⁾ In their work, five activities, including walking and running, were considered as known behaviors, and all other activities were classified as unknown activities. In addition, Guo *et al.* performed novelty detection based on human activity recognition using the inertial sensor data of an on-body device.⁽²⁰⁾ They considered six activities, including walking and running, and changed the combination of known and unknown activities. In our earlier work,⁽⁹⁾ we attempted to detect unknown on-body device positions with a single novelty detector, which we consider the first attempt to solve the on-body device localization problem. In the work, three types of supervised novelty detection methods were applied in accordance with the categorization by Domingues *et al.*:⁽²¹⁾ one-class support vector machine as a domain-based method, local outlier factor (LOF) as a density-based method, and isolation forest as an isolation-based method. We concluded that LOF is applicable to unknown position detection; however, its accuracy is insufficient and thus needs to be further improved. In this study, we attempt to improve the accuracy of unknown position detection with an ensemble of LOF-based novelty detectors.

3. Proposed Method

3.1 On-body device localization system with unknown position detection

Figure 1 illustrates the basic processing flow of the on-body device localization system. As described in Sect. 2, the novelty detector requires the knowledge of a familiar class and is trained in advance by the same data as for the recognition component. Novelty detection works as a filter to feed only the data (samples) from known positions to the position recognition component. This technique helps the component to avoid misclassifying data from an unknown position into one of n known positions. This is a benefit of novelty detection from the system viewpoint of reliability. In addition to simply rejecting the data from unknown positions, the data judged as unknown can be used to expand the supported classes of the system, i.e., known positions, by the user's active involvement in labeling and retraining the recognition component. This is another benefit of novelty detection from the viewpoint of extensibility.

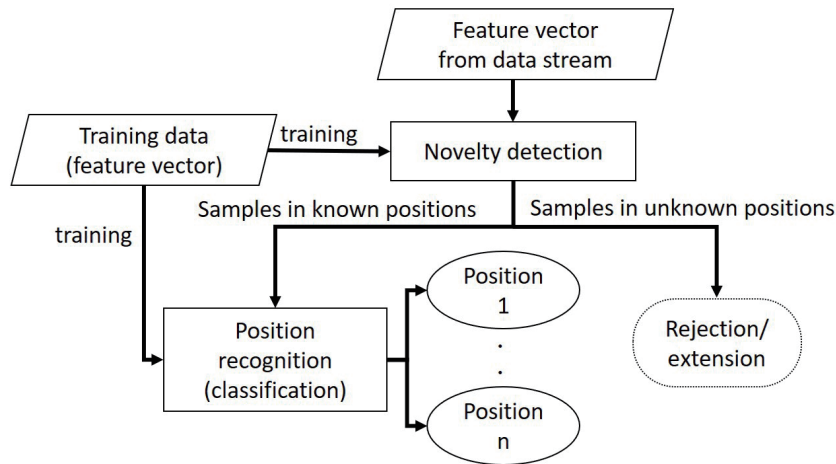


Fig. 1. Basic processing flow of on-body device localization with new position detection.

3.2 Ensemble novelty detection

As one of the better-known ensemble methods for classification and regression, random forest has achieved a high accuracy in various areas. An ensemble method makes the final decision using the results of a number of weak classifiers (regressors). In novelty detection, the effectiveness of using multiple detectors has also been demonstrated.⁽²²⁾ In this article, we propose an ensemble novelty detection method for the accurate detection of unknown storing positions. We also demonstrate a method of adjusting an important parameter that maximizes the effect of ensemble novelty detection while eliminating the burden of end users.

In random forest, randomness is introduced in two stages of building weak classifiers: the training sample and attribute (feature) subset selections. In the proposed method, we follow this approach. Figure 2 illustrates the processing flow of ensemble novelty detection. S weak detectors are trained in advance, whereby the detection features $\mathbf{F}'_{k \in \{1..S\}}$ are chosen from the original feature set \mathbf{F} ; the training data are also chosen from the original training data labeled as known by restoration extraction. In the detection phase, each detector performs novelty detection independently against a test sample (A) and the judgment of known or unknown is performed by integrating the results from each detector (B).

Generally, in ensemble classification, the final results are determined by a simple majority vote that uses the results of multiple weak classifiers. However, in the proposed ensemble novelty detector, the method that makes judgments based on the results of multiple weak detectors is not limited to a simple majority vote. Instead, we introduce a constant T called the decision threshold, and a test sample is finally judged as unknown if the number of weak detectors with the judgment of known is less than T ; otherwise, it is judged as known. In other words, S weak detectors generate S final judgment results depending on T . The majority vote is a special case with $T = S/2$. The next section presents a preliminary experiment carried out to determine the relationship between T and an evaluation parameter.

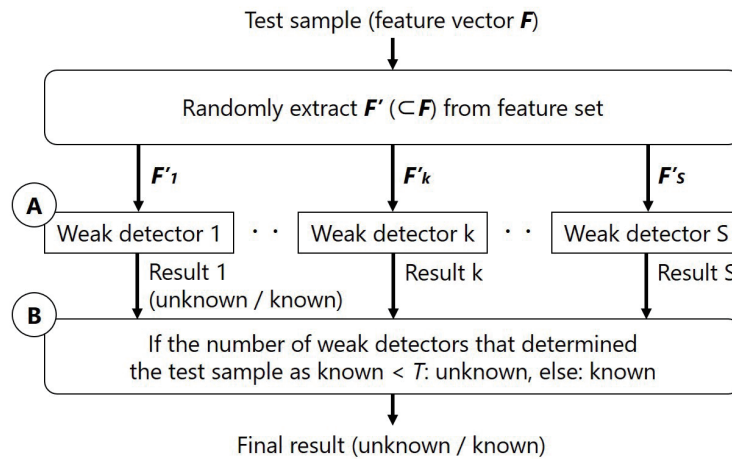


Fig. 2. Processing flow of ensemble novelty detection.

3.3 Preliminary experiments

This section presents preliminary experiments to determine the characteristics of ensemble novelty detection.

3.3.1 Dataset and preprocessing

Three datasets were collected from smartphones carried in a wide variety of ways, such as in trouser pockets, in the hand, or in the handbag (Table 2). Datasets A and B^(2,6) were collected in the authors' laboratory, whereas dataset C⁽²³⁾ is a publicly available dataset. All datasets are composed of three-axis accelerometer signals. The features designed for position recognition were used for each dataset, which include the mean and standard deviation in the time domain and the entropy and energy in the frequency domain, for example.

3.3.2 Evaluation parameter

As shown in Sect. 3.2, T contributes to novelty detection performance. Normalized accuracy (NA) was used as an evaluation parameter, which is represented by Eqs. (1)–(3). NA is the average of the true positive rate (TPR) and true negative rate (TNR). Here, the decision of unknown is assumed as positive and the decision of known is negative. Thus, the TPR indicates the ratio of samples correctly judged as unknown to the total number of unknown samples. By contrast, the TNR is represented by the ratio of samples correctly judged as known to the total number of actually known samples. Meanwhile, global accuracy (GA) is represented by the ratio of samples correctly judged as unknown and known to the total number of samples [Eq. (4)]. Compared with GA, NA is considered a more effective parameter when a difference is observed between the numbers of unknown and known samples.

Table 2
Dataset used in the preliminary experiment.

Dataset	Person	Position (underlines indicate the candidates for a known position)	# of features
A ⁽⁶⁾	70	neck (hanging), <u>chest pocket</u> , <u>jacket pocket</u> , <u>trouser front/back pockets</u> , bag (<u>backpack</u> , handbag, and shoulder bag), hand (calling, watching the screen in the portrait direction, and <u>swinging during walking</u>)	30
B ⁽²⁾	20	neck (hanging), <u>chest pocket</u> , <u>jacket pocket</u> , <u>trouser front/back pockets</u> , bag (<u>backpack</u> , handbag, shoulder bag, and messenger bag)	63
C ⁽²³⁾	10	<u>trouser left/right pockets</u> , <u>arm</u> , <u>wrist</u> , <u>belt</u>	16

$$NA = \frac{TPR + TNR}{2} \quad (1)$$

$$TPR = \frac{TP}{TP + FN} \quad (2)$$

$$TNR = \frac{TN}{TN + FP} \quad (3)$$

$$GA = \frac{TP + TN}{TP + FN + TN + FP} \quad (4)$$

3.3.3 Effect of ensemble novelty detection

We first examined the relationship between T and NA by changing T in dataset A. There were two groups of 35 subjects (70 in total). The data obtained from the subjects in the first and second groups were used as the training and test data, respectively. The average of the results obtained from the 35 test subjects was taken as the result of the experiment.

The experimental system was implemented using scikit-learn 0.20.3, a Python machine learning library. In the random forest implementation of scikit-learn, the default value for the number of features in weak classifiers is the square root of the original number of features. By following this convention, the size of feature vectors, \mathbf{F}' , randomly selected for each weak classifier is five ($\approx \sqrt{30}$). Moreover, the number of weak detectors, S , was set to 100. The fixed set of utilized known positions was composed of the following: chest pocket, jacket pocket, trouser front pocket, trouser back pocket, backpack, and swinging during walking, which are underlined in Table 2.

Figure 3 shows the change in NA when T was varied. NA_{NonEns} is the case where a single detector is used. At $T = 63$, NA in the ensemble case had a maximum value of 0.864, which is an increase of 0.186 compared with that in the case of NA_{NonEns} (0.678). In addition, NA for the majority vote ($T = 50$) was 0.818, which is 0.046 lower than the maximum value. These results suggest that the majority vote is not the best solution for the ensemble method and that NA can be greatly increased by setting an appropriate T .

Hereafter, the T with the highest NA is represented as T_{max} , whereas the T obtained by the majority vote is represented as T_{mv} . Moreover, $NA(T)$ represents NA at T . In Sect. 3.4, we describe a method of estimating T_{max} as a hyperparameter.

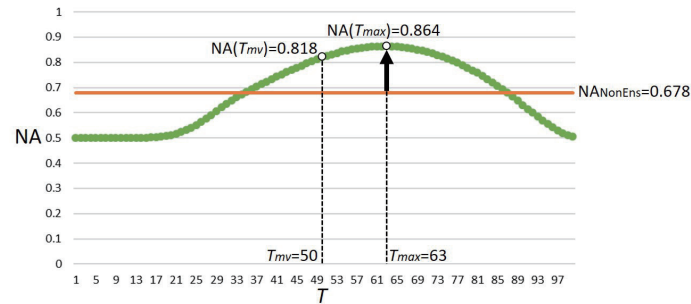


Fig. 3. (Color online) Relationship between T and NA.

3.3.4 Effect of the number of known classes

In addition, we examined the effect of the number of known classes in T_{max} by changing the combination of known positions. In addition to dataset A, datasets B and C were also used. Similarly to dataset A, the size of feature vectors for each weak detector (F') is the square root of the size of the original feature vectors $|F|$. Thus, eight and four features were randomly selected in experiments for datasets B and C, respectively.

The evaluation was carried out using a leave- p -persons-out cross-validation ($LpPO$) scheme, in which data from p persons were used to train the ensemble novelty detector and the data from the other persons were used for the test. The case with $p = 1$ is a special case called leave-one-person-out cross-validation (LOPO-CV), which is a popular evaluation method in machine learning. However, to reduce the time of evaluation, we took half of the persons in the dataset as p and performed the evaluation twice by swapping the roles of training and testing. Thus, data from 70, 20, and 10 persons were all used for training and testing for datasets A, B, and C, respectively. The average of the two tests was used as the evaluation result.

Various combinations of known positions were examined. Each dataset is divided into a group of known positions and a group of unknown positions, in which the data from known positions were used to train the ensemble novelty detector and the data from other positions were used to test the detector. As described above, we used the $LpPO$ approach, and the data for training consisted of the known positions obtained only from half of the subjects, that is, p persons in the training group, whereas the data for the test included the known and unknown positions obtained from another p persons as a realistic condition. Ideally, all possible combinations should be used as known positions in the evaluation. However, in the case of dataset A, the total number of combinations was huge, i.e., $2035 (= \sum_{i=2}^{10} {}_{11}C_i)$, where the number 11 comes from the number of positions supported in dataset A, whereas 2 and 10 are the minimum and maximum numbers of positions regarded as known, respectively. To reduce the number of possible combinations, popular storing positions were selected for known classes based on a literature survey, as shown in Table 1. These are regarded as the candidates for known positions, which are underlined in Table 2. k positions were used as known positions, and the rest were considered unknown positions. In total, $57 (= \sum_{k=2}^6 {}_6C_k)$ sets of known positions were formed for dataset A, whereas five ($11 - 6$) to nine ($11 - 2$) positions were used

as unknown positions. Similarly, in datasets B and C, 26 ($= \sum_{k=2}^5 5C_k$) and 25 ($= \sum_{k=2}^4 5C_k$) sets of known positions were formed, respectively. As the number of classes in dataset C is small, i.e., five, at most four positions were used as known.

Table 3 shows the average T_{max} for different numbers of known positions when the number of weak detectors is 100. In datasets A and B, the differences between T_{max} with k known positions and T_{max} with $(k - 1)$ known positions are within 2.0 to 3.9 and 1.8 to 4.2, respectively, which we consider small. Exceptionally, in dataset C, the difference in T_{max} between $k = 3$ and $k = 4$ is 9.1. We consider that this is due to the small number of subjects in the training dataset, i.e., five. Therefore, although exceptions are present, T_{max} obtained with k known positions and T_{max} obtained with $(k - 1)$ known positions can be regarded as close. In the next section, we present a method of estimating T_{max} based on this principle.

3.4 Algorithm for estimating T_{max}

As shown in Sect. 3.3.3, if we know T_{max} in advance, then the system can fully benefit from the ensemble novelty detection. However, in practice, unknown classes cannot be included in the dataset for evaluation. Therefore, we propose a method of estimating T_{max} using only training samples of known classes. The proposed estimation method is based on the principle shown in Sect. 3.3.4: T_{max} with k known positions is close to T_{max} with $(k - 1)$ known positions. Let k , n , and S be the number of known positions, the number of subjects, and the number of weak detectors, respectively. The algorithm used to estimate T_{max} is as follows.

- Step 1: In one weak detector, calculate NA by using the data from one subject as test data and the data from the remaining $(n - 1)$ subjects as training data, also regarding one of the positions as an unknown position and the remaining $(k - 1)$ positions as known positions. Figure 4 shows a state where Subject 1 is the test subject and the other subjects are used to train the detector. Also, Position 1 is set to the unknown position.
- Step 2: Perform Step 1 by changing the weak detector; iterate until all weak detectors have been evaluated, i.e., S times.
- Step 3: Obtain S final (integrated) results based on the S results of weak detectors by changing T , and calculate NA for each T .
- Step 4: Perform Steps 1 to 3 with a different test subject; iterate until all (n) subjects in the dataset are tested.
- Step 5: For each T , calculate the average of n NAs, and check T_{max} : T when the calculated average NA is the highest.

Table 3
Average T_{max} for different numbers of known positions ($S = 100$).

# of known positions (k)	Dataset A	Dataset B	Dataset C
2	50.8	70.7	46.2
3	54.7	74.9	48.1
4	56.9	76.7	57.2
5	59.0	78.5	—
6	61.0	—	—

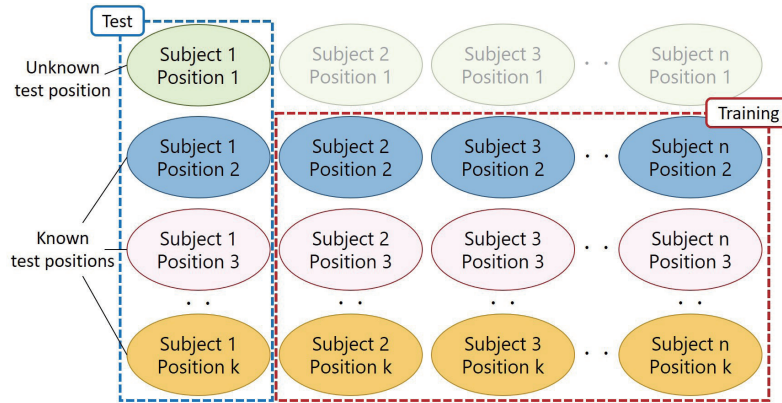


Fig. 4. (Color online) Calculate NA while changing test subjects.

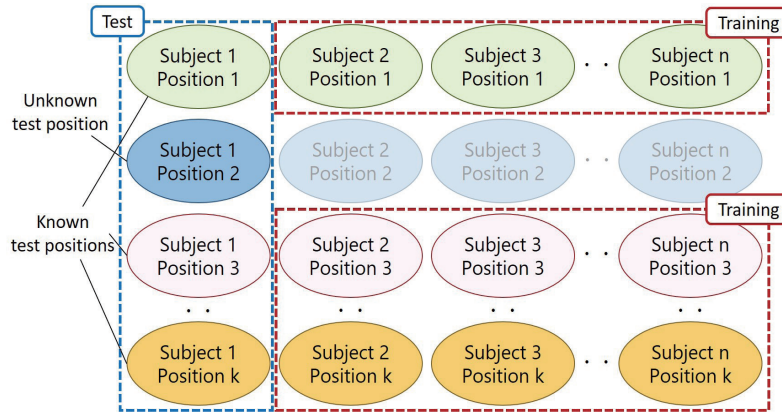


Fig. 5. (Color online) Calculate NA while changing unknown positions.

Step 6: Perform Steps 1 to 5 while changing the unknown position; iterate k times. Figure 5 shows a state where Position 2 is the unknown position.

Step 7: Calculate the average of the k values of T_{max} obtained in Step 5 and regard it as T_{max} .

T_{max} is represented as T , which corresponds to the highest average NA of all combinations of known positions calculated using the LOPO-CV scheme. The steps described above need to be performed when the number of known classes is changed. Therefore, an end user does not need to do anything unless a new storing position is registered to the recognition targets during use. We discriminate the estimated T_{max} from the true T_{max} by referring to the estimated one as T_{est} .

4. Evaluation Methodology

The proposed ensemble novelty detection method was evaluated using the three datasets in more depth. This section provides a description of the method, followed by the results and discussion in Sect. 5. Unless noted otherwise, the number of weak detectors (S) and the contamination ($cont$), a parameter in LOF, were set to 100 and 0.05, respectively.

4.1 Basic performance of the proposed method

The basic detection performance of the proposed method was evaluated using the three datasets shown in Table 2. The evaluation was carried out in the same manner as in Sect. 3.3.4. Here, $NA(T_{est})$ was also calculated.

4.2 Effectiveness against parameter variations

The effectiveness of the ensemble novelty detection and the T_{max} estimation method was evaluated in depth against two parameters: the number of weak novelty detectors (S) and the contamination parameter in the LOF-based novelty detector.

Generally, in ensemble classification, accuracy increases with the number of weak classifiers and converges at a certain level.⁽⁶⁾ Thus, the number of weak detectors was changed to 20 and 50 to confirm this tendency in ensemble novelty detection. At the same time, the feasibility of the ensemble novelty detection and the proposed T_{max} estimation method for various numbers of weak detectors was evaluated.

Furthermore, in LOF, $cont$ is an important parameter that represents the novelty threshold. If $cont$ increases, then the test sample is more likely to be judged as unknown. Here, we set the value to 0.01 and 0.10 to verify the effectiveness of the proposed method for various $cont$ values.

5. Results and Discussion

5.1 Effectiveness of the proposed estimation method

Although the experiment in Sect. 4.1 covered all possible combinations of known classes in dataset A, here we pick a case that we used in the preliminary experiment in Sect. 3.3.3 to examine the accuracy of T_{est} estimation in a concrete example. In this case, the combination of known positions consists of the following: chest pocket, jacket pocket, trouser front pocket, trouser back pocket, backpack, and swinging during walking. In Fig. 6, we add T_{est} and $NA(T_{est})$ to Fig. 3 to compare different T values and the corresponding NAs. The figure shows that T_{est} is much closer to T_{max} than T_{mv} , which indicates that the proposed estimation method performed better than the mere majority vote in this case.

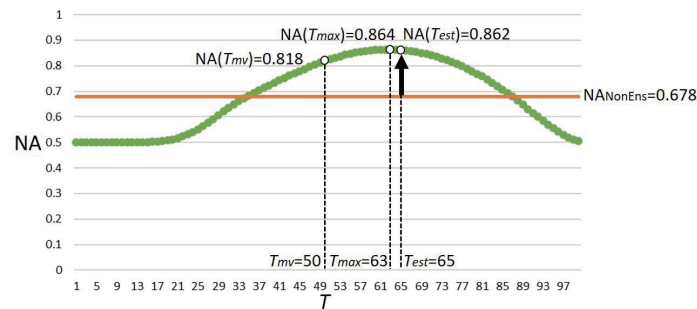


Fig. 6. (Color online) Various T values and corresponding NAs under the same conditions as in Sect. 3.3.3.

Figure 7 shows the overall experimental results, i.e., the averages of 57, 26, and 25 combinations of known positions for datasets A, B, and C, respectively. A similar tendency can be found in datasets A and B, indicating that $NA(T_{est})$ is closer to $NA(T_{max})$ than $NA(T_{mv})$ and (NA_{NonEns}). However, in dataset C, $NA(T_{mv})$ is larger than $NA(T_{est})$. This is because the number of subjects in the training data was too small (5) to construct the novelty model well. The results discussed in Sect. 3.3.4 also show that the underlying assumption required for applying the T_{est} estimation method was not established. For the other two datasets, a decision threshold that is close to the ideal one (T_{max}) can be successfully estimated from the data from persons whose data are not used for training the novelty detectors.

5.2 Effects of features on ensemble novelty detection

Similarly to a general classification task, the number and types of features are considered to affect the detection performance of novelty detection. In this study, different features were used for each of the three datasets; however, a commonly effective set of features may exist for the same task, i.e., on-body device position recognition. Finding such a feature set is an agenda for future work. Meanwhile, the proposed ensemble method detects unknown classes for each weak detector using different subset of features, which means that novelty models are generated in a different feature spaces. In general, a feature set that is not suitable for detecting any unknown class is difficult to identify in advance. A single detector case (NonEns) may fail to detect a novelty class with the particular set of features. By contrast, in a collection of weak detectors working in heterogeneous feature spaces, some of them performs correctly and others do not. If the number of correctly performed detectors is above the decision threshold, an overall detection is correct. We consider that this is a reason that the ensemble methods using T_{mv} , T_{est} , and T_{max} showed better NAs than NA_{NonEns} as shown in Fig. 7.

5.3 Robustness against the number of weak detectors

Figure 8 shows the results obtained when the numbers of weak detectors (S) are 20, 50, and 100. As S increases, so does $NA(T_{max})$; the rate of increase is low, indicating a tendency

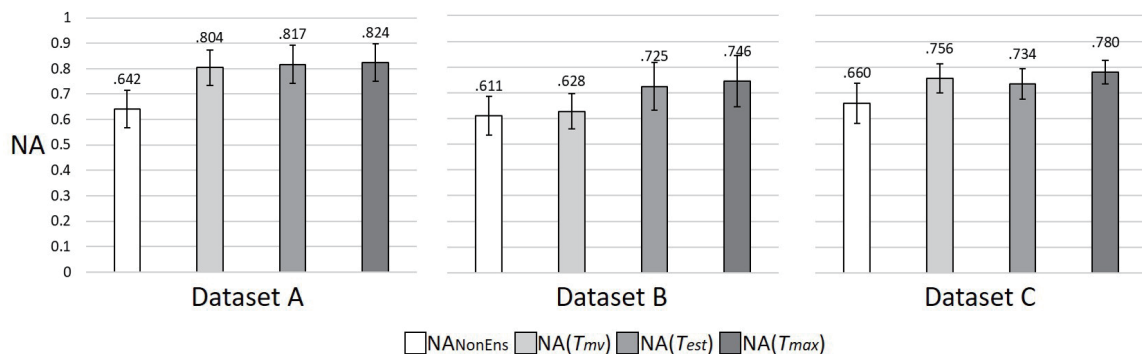


Fig. 7. NAs with various decision thresholds in the ensemble method as well as in the non-ensemble method. The ranges indicate the standard deviations.

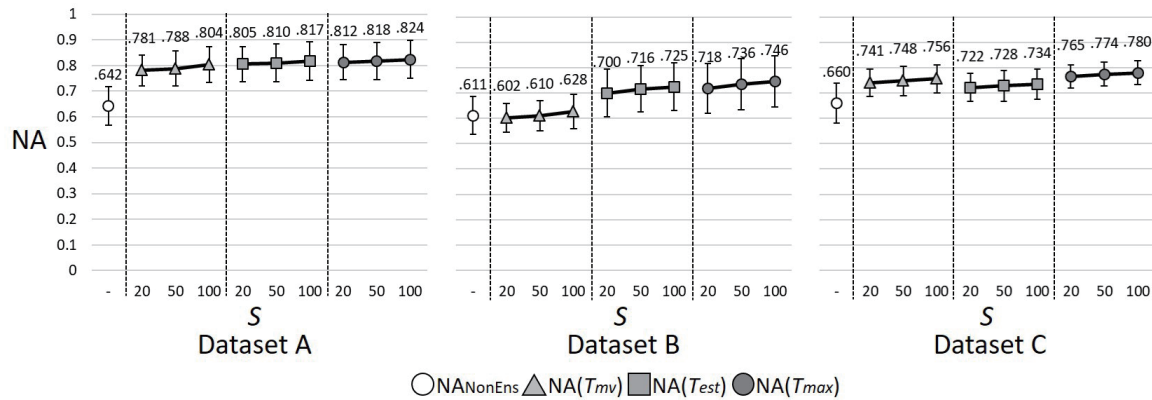


Fig. 8. Relationship between the number of weak detectors (S) and NA with various decision thresholds in the ensemble method as well as in the non-ensemble method. The ranges indicate the standard deviations.

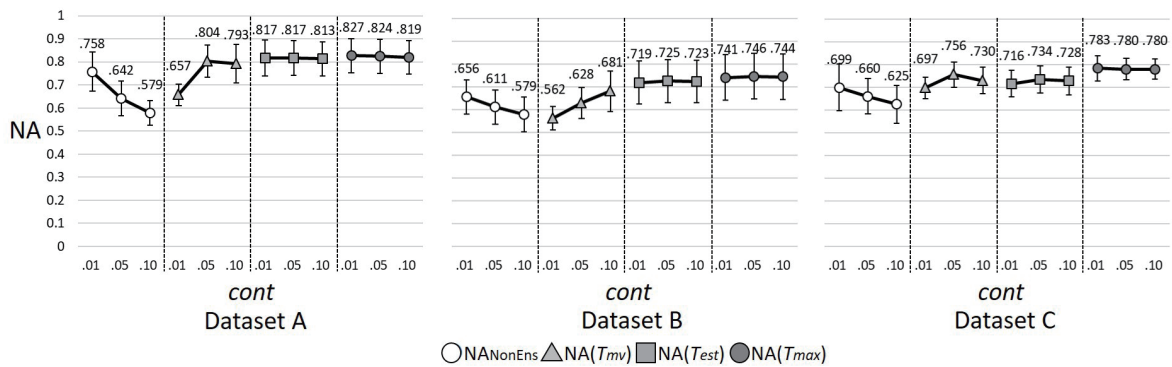


Fig. 9. Relationship between $cont$ and NA with various decision thresholds in the ensemble method as well as in the non-ensemble method. The ranges indicate the standard deviations.

similar to that of the general ensemble classification. Moreover, $NA(T_{est})$ in datasets A and B is closer to $NA(T_{max})$ than $NA(T_{mv})$ and also much higher than NA_{NonEns} . $NA(T_{est})$ in dataset C also increases with S ; however, it is lower than $NA(T_{mv})$ as discussed in the previous section. Therefore, the proposed estimation method is applicable even if the number of weak detectors is changed; however, there is a case in which $NA(T_{est})$ is lower than $NA(T_{mv})$.

As shown above, $NA(T_{est})$ increases with the number of weak detectors; however, the increase in $NA(T_{est})$ can also increase the requirements for computational memory space and processing speed. Thus, a trade-off exists between the number of weak detectors and these requirements. Therefore, an appropriate number of weak detectors should be determined depending on the desired NA and available computational power.

5.4 Robustness against novelty threshold (contamination)

Figure 9 shows the results obtained when the $cont$ values are 0.01, 0.05, and 0.10. As shown in this figure, NA_{NonEns} markedly changes with $cont$, whereas the variations in $NA(T_{max})$ and

$NA(T_{est})$ are not as large as that in NA_{NonEns} . This finding suggests that the ensemble novelty detection is less affected by *cont*. Furthermore, the calculated $NA(T_{est})$ is closer to $NA(T_{max})$ than $NA(T_{mv})$ for all *cont* values for datasets A and B and in the case of *cont* = 0.01 for dataset C. Therefore, the proposed estimation method is effective in most cases.

The important point is that the effects of the proposed ensemble novelty detection and T_{est} estimation are reasonably independent of the internal parameter of the LOF-based single novelty detector, which implies that the adjustment of *cont* is not such a critical task.

6. Conclusion

We proposed an ensemble novelty detection method and applied it to the on-body device position recognition problem. This method integrates the results from multiple weak novelty detectors to infer whether a device is located in a known or unknown position. We devised a method of estimating the decision threshold for the known position that maximizes the overall accuracy of novelty detection (T_{max}). This method was evaluated using three datasets while changing parameters such as the combination of positions regarded as known, the number of weak detectors, and the hyperparameter of the LOF-based novelty detector, i.e., contamination. For various conditions, we confirmed that the ensemble novelty detection with the estimated decision threshold outperformed a single novelty detector in all three datasets. The accuracy obtained with the proposed T_{max} estimation (T_{est}) was comparable to that obtained with T_{max} and was more effective than a mere majority vote in two datasets; however, one of the datasets with a relatively small number of persons (five) showed a lower accuracy than the results with the majority vote. We also confirmed that the detection accuracy increased with the number of weak detector (S), while there was little change in the accuracy with respect to contamination (*cont*). Thus, S is the dominant parameter in the proposed method.

We consider that the proposed estimation method is applicable to a wide variety of novelty detection problems in which the training data are obtained from multiple subjects and constitute multiple classes in addition to on-body position recognition, such as human daily activity recognition and fall detection. In the future, the applicability will be verified in such domains.

Acknowledgments

This work was funded by the Japan Society for the Promotion of Science (Grant No. 18H03228).

References

- 1 Y. Cui, J. Chipchase, and F. Ichikawa: 12th Human-Computer Interaction (HCI International, 2007) 483.
- 2 K. Fujinami: Information 7 (2016) 21.
- 3 K. Fujinami: Int. J. Smart Sens. Intell. Syst. 9 (2016) 2257.
- 4 A. Exler, C. Dinse, Z. Günes, N. Hammoud, S. Mattes, and M. Beigl: 2017 ACM Int. Joint Conf. Pervasive and Ubiquitous Computing (UbiComp' 17) 970.
- 5 A. Mannini, A. M. Sabatini, and S. S. Intille: Pervasive Mob. Comput. 21 (2015) 62.
- 6 K. Fujinami, T. Saeki, Y. Li, T. Ishikawa, T. Jimbo, D. Nagase, and K. Sato: Int. J. Adv. Comput. Sci. Appl. 8 (2017) 447.

- 7 K. Alanezi and S. Mishra: *Comput. Electr. Eng.* **44** (2015) 307.
- 8 O. D. Incel: *Sensors* **15** (2015) 25474.
- 9 M. Saito and K. Fujinami: 2019 IEEE 8th Global Conf. Consumer Electronics (GCCE, 2019).
- 10 L. Breiman: *Mach. Learn.* **45** (2001) 5.
- 11 Y. Shi, Y. Shi, and J. Liu: 13th Int. Conf. Ubiquitous Computing (UbiComp' 11) 559.
- 12 M. Bieshaar: arXiv e-prints (2018). <https://arxiv.org/abs/1803.02097>
- 13 T. Sztyley and H. Stuckenschmidt: 2016 IEEE Int. Conf. Pervasive Computing and Communications (PerCom, 2016) 1.
- 14 J. Wiese, T. S. Saponas, and A. J. B. Brush: 2013 SIGCHI Conf. Human Factors in Computing Systems (CHI, 2013).
- 15 R. Yang and B. Wang: *Information* **7** (2016) 72.
- 16 J. A. Aghamaleki and S. M. Baharlou: *Expert Syst. Appl.* **105** (2018) 221.
- 17 L. Manevitz and M. Yousef: *Neurocomputing* **70** (2007) 1466.
- 18 D.-Y. Yeung and Y. Ding: *Pattern Recognit.* **36** (2003) 229.
- 19 J. Yin, Q. Yang, and J. J. Pan: *IEEE Trans. Knowl. Data. Eng.* **20** (2008) 1082.
- 20 J. Guo, X. Zhou, Y. Sun, G. Ping, G. Zhao, and Z. Li: *J. Med. Syst.* **40** (2016) 140.
- 21 R. Domingues, M. Filippone, P. Michiardi, and J. Zouaoui: *Pattern Recognit.* **74** (2018) 406.
- 22 A. AlSuwaidi, B. Grieve, and H. Yin: *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **11** (2018) 1041.
- 23 M. Shoaib, S. Bosch, O. D. Incel, H. Scholten, and P. J. M. Havinga: *Sensors* **14** (2014) 10146.

About the Authors

Mitsuaki Saito received his B.S. degree in computer science from Tokyo University of Agriculture and Technology (TUAT), Japan, in 2019 and is currently an M.S. student at TUAT. His research interests are machine learning and activity recognition. (perorokekey@gmail.com)

Kaori Fujinami received his B.S. and M.S. degrees in electrical engineering and his Ph.D. in computer science from Waseda University, Japan, in 1993, 1995, and 2005, respectively. From 2005 to 2006, he was a visiting lecturer at Waseda University. From 2007 to 2017, he was an associate professor at the Department of Computer and Information Sciences at TUAT. In 2018, he became a professor at TUAT. His research interests are machine learning, activity recognition, human–computer interaction, and ubiquitous computing. (fujinami@cc.tuat.ac.jp)