Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Man-machine verification of mouse trajectory based on the random forest model*

Zhen-yi XU[1], Yu KANG‡[1,2], Yang CAO[1], Yu-xiao YANG[1]

[1]*Department of Automation, School of Information Science and Technology, University of Science and Technology of China, Hefei 230022, China*

[2]*State Key Laboratory of Fire Science, University of Science and Technology of China, Hefei 230027, China*

E-mail: xuzhenyi@mail.ustc.edu.cn; kangduyu@ustc.edu.cn; forrest@ustc.edu.cn; yyx531@mail.ustc.edu.cn

Received July 4, 2017; Revision accepted Dec. 25, 2017; Crosschecked July 3, 2019

Abstract: Identifying code has been widely used in man-machine verification to maintain network security. The challenge in engaging man-machine verification involves the correct classification of man and machine tracks. In this study, we propose a random forest (RF) model for man-machine verification based on the mouse movement trajectory dataset. We also compare the RF model with the baseline models (logistic regression and support vector machine) based on performance metrics such as precision, recall, false positive rates, false negative rates, F-measure, and weighted accuracy. The performance metrics of the RF model exceed those of the baseline models.

Key words: Man-machine verification; Random forest; Support vector machine; Logistic regression; Performance metrics

https://doi.org/10.1631/FITEE.1700442 **CLC number:** TP181

1 Introduction

The security landscape is changing with internal threats and financial motivations, replacing the activities of "script kiddies," and seeking bragging rights about the number of machines compromised (Gordon et al., 2006). Unauthorized access, theft of proprietary information, and insider net abuse are among the top five financial losses. Fig. 1 shows that the decline in the overall frequency of (successful) misuses of computer systems, which began in 2001, may have come to a halt in 2005. The percentage of respondents answering that their organization experienced unauthorized use of computer systems in the last 12 months increased slightly from 53% in 2004 to 56% in 2005. Furthermore, the percentage of respondents answering that there was no unauthorized use of their organization's computer systems decreased from 35% to 31%. Many respondents said that they did not know if unauthorized access had occurred, and the percentage of these respondents increased from 11% to 13% (Brown and Rogers, 1993).

Man-machine verification in this study is based on behavioral verification technology, which breaks through the one-dimensional protection of the "computer graphics" of code verification technology. The concept of "behavior" is introduced in the verification of security technology, which incorporates many disciplines including biology, interaction psychology, and artificial intelligence. The "behavioral identification module" is used to generate and analyze multiple behavioral data. After multi-processing in the "sandbox module" in real time to make accurate judgments, a multi-complex and different dimensional high-intensity defense system can be established. In the behavioral verification application, the validation

‡ Corresponding author

* Project supported by the National Natural Science Foundation of China (Nos. 61673361 and 61422307)

ORCID: Zhen-yi XU, http://orcid.org/0000-0002-5804-882X

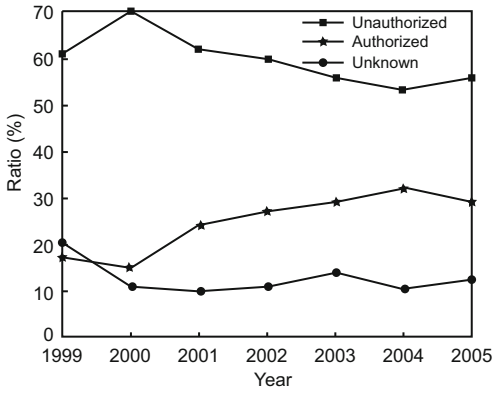

Fig. 1 Different use of computer systems from 1999 to 2005

area is fixed. Some malicious applications can easily simulate the mouse drag action and complete the verification. The challenge in engaging man-machine verification involves the correct classification of man and machine tracks.

Weiss et al. (2007) tested the feasibility of mouse movement biometrics by k-nearest neighbors (KNN). Su (2016) proposed a gradient boosting decision tree (GBDT) classification model on the behavior-based verification code, showing the potential of this method to be used in man-machine verification.

In this study, we constructe a random forest (RF) model to effectively classify and segregate man and machine tracks, based on the mouse movement trajectory dataset. We compare the prediction performance with logistic regression (LR) and support vector machine (SVM) on both the training and testing data sets. We show that the proposed RF model has favorable prediction performance. Our efforts can be summarized as follows: (1) We adapt the RF classification model to handle the task of man-machine verification based on mouse trajectory data. (2) We describe feature definitions and create a feature vector, which represents the characteristics of the movement tracks. (3) We compare the prediction performance of the proposed RF model with those of LR and SVM on both the training and testing data sets.

2 Dataset

We use the data in our experiment to extract dynamic characteristics from interactive sliding verification code track data. The data are acquired from a network technology company. When any user logged in or registered on the web site by using the drag operation in completely automated public turing test to tell computers and humans apart (CAPTCHA) to complete the authentication, the server automatically recorded the horizontal, vertical, and time statistics.

In this study, we evaluate our method on a dataset consisting of 3000 records (2600 positive samples and 400 negative samples) of different mouse trajectories. Each row in the records is defined in Table 1. An example of the records is as follows: {5;241,2594, 904; 276, 2555, 2527;297, 2542, 2593;360, 2555, 5284; 1574.5, 267; 0}.

Table 1 Definition of the mouse trajectory records

Field	Type	Explanation
a1	bigint	Record number
a2	string	Mouse trajectory (x_t,y_t,t)
a3	string	Target position (x_f,y_f)
label	string	1: normal track; 0: machine track

3 Methodology

3.1 Data preparation

Through the data samples, we can see that the collected mouse trajectory records are in time sequence. Therefore, the mouse movement speed, movement offset, movement interval, and movement slope can be calculated from the mouse tracks. Definitions of these statistics are as follows, which make up the feature vectors (Taieb and Hyndman, 2014).

Definition 1 (Movement speed) In the human-computer interaction process, the mouse movement of the track is usually not a straight line: the mouse directions of movement and vertical movement are shifted. The velocity direction of a single point is not always consistent with the direction of movement, and sometimes there is even a great deviation. The average speed of the curve is defined as the average speed taken to complete the curve. The speed between two points is computed as the distance traveled over time. The speed at the single point v is decomposed into the speed along the moving direction and the speed along the vertical moving direction. For horizontal movement, the velocity in the horizontal direction is calculated and the speed at the j^{th} data point is defined as the difference between the j^{th} data point and the horizontal

coordinates of the $(j-1)^{\text{th}}$ data point divided by their time stamps. The vertical velocity can also be calculated. For each movement segment, the mean and variance of vertical and horizontal speeds can be calculated, denoted separately as $v_{x\text{mean}}$, $v_{x\text{var}}$, $v_{y\text{mean}}$, and $v_{y\text{var}}$. The speed feature is defined as $\boldsymbol{f}_{\text{speed}} = [v_{x\text{mean}}, v_{x\text{var}}, v_{y\text{mean}}, v_{y\text{var}}]$.

Definition 2 (Movement offset) A movement offset includes horizontal movement and vertical movement. For horizontal movement, the offset in the vertical direction is calculated, and the offset of the j^{th} movement is defined as the sum of the absolute values of the difference in the vertical coordinates of each adjacent data point in the moving trajectory. For vertical movement, the horizontal offset is calculated. For each movement segment, the mean and variance of the vertical and horizontal offsets can be calculated, denoted separately as x_{mean}, x_{var}, y_{mean}, and y_{var}. The movement offset feature is defined as $\boldsymbol{f}_{\text{offset}} = [x_{\text{mean}}, x_{\text{var}}, y_{\text{mean}}, y_{\text{var}}]$.

Definition 3 (Movement interval) For each movement segment, we calculate the mean and variance of the time interval of adjacent data points in the moving trajectory, denoted separately as t_{mean} and t_{var}. The movement interval feature is defined as $\boldsymbol{f}_{\text{time}} = [t_{\text{mean}}, t_{\text{var}}]$.

The feature vector is $[\boldsymbol{f}_{\text{speed}}, \boldsymbol{f}_{\text{offset}}, \boldsymbol{f}_{\text{time}}]$.

3.2 Random forest

RF is a multiple decision tree classifier based on the classification and regression tree (De'ath and Fabricius, 2000). For each tree, this method performs bootstrap sampling and enables the calculation of an error estimate based on the instances remaining "out-of-bag." RF, unlike classification and regression tree (CART), does not consider all variables at each node to determine the best split threshold, but a random subset of the original set of features. The number of variables per node is typically set to the square root of the total number of variables, but it can be adjusted by the user. Those two mechanisms, sampling and using random variables for each node, create very different uncorrelated trees. Another user-defined parameter is the number of trees, which must be sufficiently large to capture the full variability of the training data and yield good classification accuracy. RF assigns the final class to an object based on the majority vote of all trees in the forest.

3.3 Classification

In man-machine verification research, machine learning methods have been used such as support vector machines (SVM), logistic regression (LR), and artificial neural networks (ANN) (Sanjaa and Chuluun, 2013). In this study, we use RF as a feasible method for man-machine verification.

In terms of supervised learning, various studies have ranked gradient boosted trees, random forests, neural networks, and support vector machines as having high predictive accuracies (Caruana and Niculescu-Mizil, 2006; Liu et al., 2015). Although gradient boosted trees have the highest accuracy, RF is able to achieve almost the same performance with minor parameter tuning (Hultquist et al., 2014).

In this study, we implement RF on Python using the random forest library. The RF classifier model takes feature vector $[\boldsymbol{f}_{\text{speed}}, \boldsymbol{f}_{\text{offset}}, \boldsymbol{f}_{\text{time}}]$ as input. The feature vector is then normalized to create a normalized feature vector in the form of Eq. (1):

$$\overline{x} = \frac{x - x_{\min}}{x_{\max} - x_{\min}}, \tag{1}$$

where $x_{\min}$ and $x_{\max}$ are the minimum and maximum values of the measurement over all samples, respectively. Normalization provides measurement values in the range 0–1 to give each measurement a roughly equal weight.

3.4 Evaluation criteria

In an imbalanced learning dataset (Chen et al., 2004), the overall classification accuracy is not an appropriate measure of performance. We use true positive rate (TPR), true negative rate (TNR), weighted accuracy (w-Acc), G-mean, Precision, Recall, and F-measure to evaluate the prediction model. All the metrics are functions of the parameters in Table 2. The performance metrics are defined as

$$\text{TNR} = \frac{\text{TN}}{\text{TN} + \text{FP}}, \tag{2}$$

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{3}$$

$$G\text{-mean} = \sqrt{\text{TNR} \times \text{TPR}}, \tag{4}$$

$$\text{w-Acc} = \beta \cdot \text{TPR} + (1-\beta)\text{TNR}, \tag{5}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{6}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} = \text{TPR}, \tag{7}$$

$$F\text{-measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}, \tag{8}$$

where β and $(1-\beta)$ are the class weights of the positive and negative classes, respectively.

Table 2 Confusion matrix

Actual value	Prediction outcome	
	Retrieved	Not retrieved
Relevant	True positive (TP)	False negative (FN)
Non-relevant	False positive (FP)	True negative (TN)

We also use the receiver operating characteristic (ROC) curve and the area under curve (AUC) to show the superiority of the proposed RF model to other classification models.

4 Experimental results

The proposed method was implemented in a personal computer with an Intel Core i3-3220 3.30 GHz processor. In the experiments, the training set consisted of 12 feature vectors with corresponding labels. To validate the RF model, we compared it with the following baselines: (1) linear regression, which is applied for each record individually and is a single-task learning method; (2) support vector machine, which is an algorithm that can classify records linearly or nonlinearly.

The importance of features is ranked as shown in Table 3. Apparently, the movement speed and movement offset features make larger contributions to the classification model. Fig. 2 shows the correlation matrix between the selected features and trajectory labels, where each row or column denotes one feature. Fig. 3 ranks the importance of features in Table 3.

Table 3 Feature importance ranking

Index	Feature	Importance
0	t_{mean}	0.094 655
1	t_{var}	0.054 040
2	$v_{x\text{mean}}$	0.048 928
3	$v_{x\text{var}}$	0.083 988
4	x_{mean}	0.210 774
5	x_{var}	0.143 252
6	$v_{y\text{mean}}$	0.041 525
7	$v_{y\text{var}}$	0.078 242
8	y_{mean}	0.231 997
9	y_{var}	0.012 600

First, we compare the RF model with the baseline models on the training set. The results are shown in Table 4. Apparently, both RF and SVM models show great improvement over LR based on all metrics (TPR, TNR, Precision, F-measure, G-mean, and w-Acc).

The RF model was then tested on the reserved testing set consisting of the remaining 20% of the

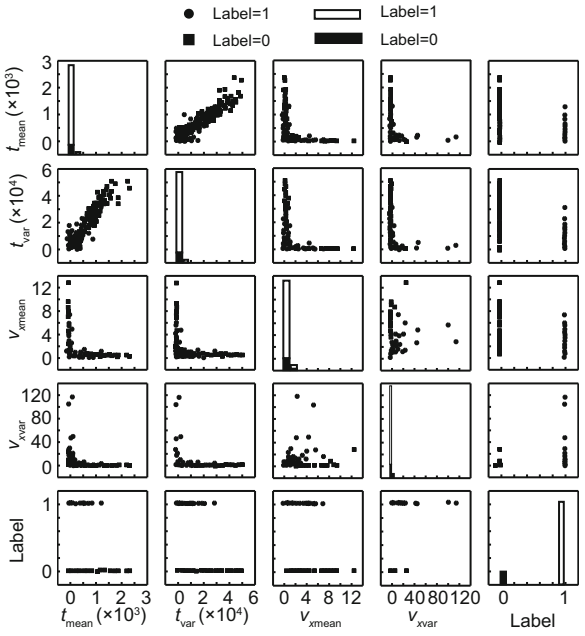

Fig. 2 Feature correlation

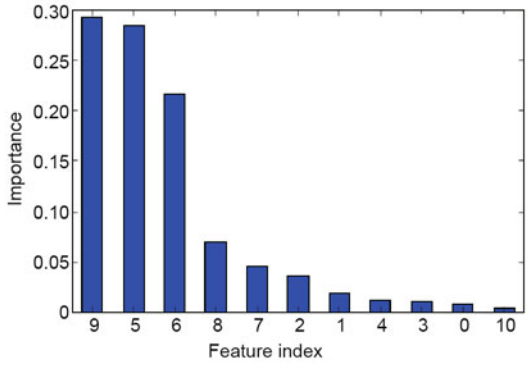

Fig. 3 Variable importance

Table 4 Performance comparison on testing set

Model	TPR	TNR	Precision	F-measure	G-mean	w-Acc
LR	0.9960	0.4271	0.9013	0.9463	0.6522	0.7116
SVM	1.0000	0.1042	0.8542	0.9214	0.3227	0.5521
RF	0.9960	1.0000	1.0000	0.9980	0.9980	0.9980

LR: logistic regression; SVM: support vector machine; RF: random forest; TPR: true positive rate; TNR: true negative rate; w-Acc: weighted accuracy

data that were not used during the training phase. Table 4 summarizes the performance comparison of the models on the testing set (Fig. 4). The RF model is superior to LR and SVM in terms of TNR, Precision, *F*-measure, *G*-mean, and w-Acc, but is worse than SVM in TPR. Fig. 5 compares the RF model and the baseline models using ROC curves on the testing dataset. From this figure, we can see that the ROC performance of the RF model is better than those of SVM and LR.

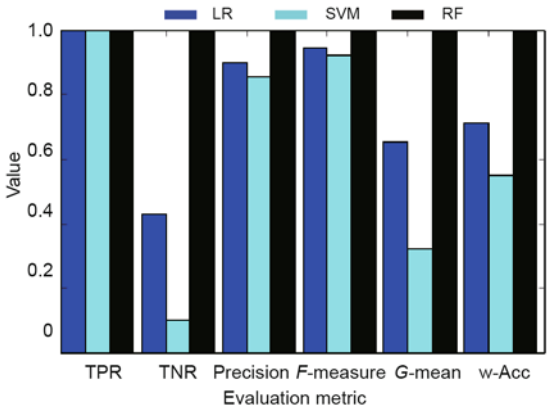

Fig. 4 Performance comparison

LR: logistic regression; SVM: support vector machine; RF: random forest; TPR: true positive rate; TNR: true negative rate; w-Acc: weighted accuracy

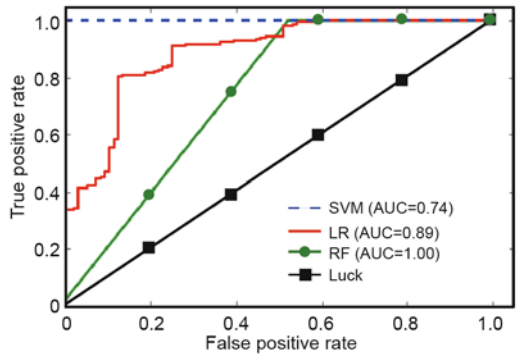

Fig. 5 ROC curve

LR: logistic regression; SVM: support vector machine; RF: random forest; ROC: receiver operating characteristic; AUC: area under curve

5 Conclusions

In this study, we constructed a random forest model for man-machine verification based on the movement speed feature $\boldsymbol{f}_{\text{speed}}$, movement offset feature $\boldsymbol{f}_{\text{offset}}$, and movement interval feature $\boldsymbol{f}_{\text{time}}$ from the mouse movement trajectory dataset. The resulting accuracies showed that the proposed random forest model achieves a 100% classification accuracy for the given dataset. Moreover, we compared the RF model and the baseline models (LR and SVM) based on different metrics: TPR, TNR, Precision, *F*-measure, *G*-mean, and weighted accuracy. The performance of the RF model was better than that of the SVM or LR model. The random forest model was used for man-machine verification. The future work on this topic can use deep learning models such as RNN or LSTM, which can provide additional insights and better training procedures for the classification model.

Compliance with ethics guidelines

Zhen-yi XU, Yu KANG, Yang CAO, and Yu-xiao YANG declare that they have no conflict of interest.

References

Brown M, Rogers SJ, 1993. User identification via keystroke characteristics of typed names using neural networks. *Int J Man-Mach Stud*, 39(6):999-1014. https://doi.org/10.1006/imms.1993.1092

Caruana R, Niculescu-Mizil A, 2006. An empirical comparison of supervised learning algorithms. 23rd Int Conf on Machine Learning, p.161-168. https://doi.org/10.1145/1143844.1143865

Chen C, Liaw A, Breiman L, 2004. Using Random Forest to Learn Imbalanced Data. Technical Report No. 666, University of California, Berkeley.

De'ath G, Fabricius KE, 2000. Classification and regression trees: a powerful yet simple technique for ecological data analysis. *Ecology*, 81(11):3178-3192. https://doi.org/10.1890/0012-9658(2000)081[3178:CARTAP]2.0.CO;2

Gordon LA, Loeb MP, Lucyshyn W, et al., 2006. 2006 CSI/FBI Computer Crime and Security Survey. Computer Security Institute, USA.

Hultquist C, Chen G, Zhao KG, 2014. A comparison of Gaussian process regression, random forests and support vector regression for burn severity assessment in diseased forests. *Remote Sens Lett*, 5(8):723-732. https://doi.org/10.1080/2150704X.2014.963733

Liu L, Yang AL, Zhou WJ, et al., 2015. Robust dataset classification approach based on neighbor searching and kernel fuzzy c-means. *IEEE/CAA J Autom Sin*, 2(3):235-247. https://doi.org/10.1109/JAS.2015.7152657

Sanjaa B, Chuluun E, 2013. Malware detection using linear SVM. 8th Int Forum on Strategic Technology, p.136-138. https://doi.org/10.1109/ifost.2013.6616872

Su T, 2016. Application of CAPTCHA with Behavioral Vilification Based on GBDT. MS Thesis, Central China Normal University, Wuhan, China (in Chinese).

Taieb SB, Hyndman RJ, 2014. A gradient boosting approach to the Kaggle load forecasting competition. *Int J Forecast*, 30(2):382-394. https://doi.org/10.1016/j.ijforecast.2013.07.005

Weiss A, Ramapanicker A, Shah P, et al., 2007. Mouse movements biometric identification: a feasibility study. Proc Student/Faculty Research Day.