algorithms

Fingerprints Classification through Image Analysis and Machine Learning Method

Huong Thu Nguyen [1,2,*] and Long The Nguyen [3]

[1] Baikal School of BRICS, Irkutsk National Research Technical University, Irkutsk 664074, Russia
[2] University of Information and Communication Technology, Thai Nguyen University, 24000 Thai Nguyen, Viet Nam
[3] Artificial Intelligence Laboratory, Institute of Information Technology and Data Science, Irkutsk National Research Technical University, Irkutsk 664074, Russia; thelong@istu.edu
[*] Correspondence: thuhuongyb@gmail.com; Tel.: +79-2460-393-41

Abstract: The system that automatically identifies the anthropometric fingerprint is one of the systems that interact directly with the user, which every day will be provided with a diverse database. This requires the system to be optimized to handle the process to meet the needs of users such as fast processing time, almost absolute accuracy, no errors in the real process. Therefore, in this paper, we propose the application of machine learning methods to develop fingerprint classification algorithms based on the singularity feature. The goal of the paper is to reduce the number of comparisons in automatic fingerprint recognition systems with large databases. The combination of using computer vision algorithms in the image pre-processing stage increases the calculation time, improves the quality of the input images, making the process of feature extraction highly effective and the classification process fast and accurate. The classification results on 3 datasets with the criteria for Precision, Recall, Accuracy evaluation and ROC analysis of algorithms show that the Random Forest (RF) algorithm has the best accuracy ($\geq$96.75%) on all 3 databases, Support Vector Machine (SVM) has the best results ($\geq$95.5%) 2/3 databases.

Keywords: fingerprint classification; singularity feature; image pre-processing; random forest; support vector Machine; machine learning

---

1. Introduction

Recently, machine learning algorithms [1–3] have outpaced previous approaches in many problems in object classification [4], object tracking [5] and image segmentation problems [6]. Practical problems in energy systems [7] that require extremely high accuracy and very low system errors are thoroughly solved by machine learning. The fingerprint classification system is also one of the systems that has a strict requirements for this. Despite of the progress made in the use of machine learning for object classification or object tracking tasks, their use in real-world applications is error-prone particularly on quality degraded input images, which is mostly human error. Almost all fingerprint images are not of perfect quality. They can be deformed, damaged by noise, such as the skin properties of each object (temperature, the humidity of the skin) are collected fingerprints, position on the finger (fingerprint angle, fingerprint area), pressure (too strong or too light) when rolling out to sample. In this study, we apply computer vision techniques [8] to improve image processing quality, focusing on noise filtering, increasing edge information. At the same time, applying machine learning algorithms, namely Random Forest algorithm [9] and Support Vector Machine (SVM) algorithm [10] to classify fingerprints into 3 types: arch, loop, whorl. The purpose is to increase the performance of the automatic fingerprint recognition system.

2. Related Work

Machine learning is increasingly proving its important in areas of practical application [11]. Therefore, scientists are researching and developing machine learning methods that are becoming increasingly optimal. One of the applications that is attracting growth is fingerprint classification. An approach to the fingerprint classification problem is researched in Reference [12]. The fingerprint is segmented into regions, this work will reduce the variance of the element directions. A relation graph is build based on the segmentation of the directional image. A model graph is used to compare with the obtained graph, can be adapted to graph matching techniques. Reference [13] presented a fingerprint classification system and its performance in an identification system used a fuzzy-neural network classifier. The classification scheme is based on fingerprint feature extraction, which involves encoding the singular points together with their relative positions and directions obtained from a binaries fingerprint image. Image analysis is carried in four stages, namely segmentation, directional image estimation, singular-point extraction and feature encoding. A method is proposed in Reference [14] based on analysis singularities and ridges relating singular points. Because of low quality images, it is very difficult to get correct positions of singular features. The authors used analysis ridge tracing and curves features to classify fingerprints. A machine learning algorithm that takes a robust approach to the classification of fingerprints is SVM. This method has created a highly accurate classification system. The advantage of the algorithm is shown in the classification of fingerprints into multiple classes. The authors of Reference [15] used a combination of the SVM algorithm with the naive Bayes method to classify fingerprints based on the number of core and delta points on fingerprints. Many researchers have tried to extract singular points in the flow of the ridges [16]. The authors of References [17–19] proposed a heuristic algorithm with singularities to classify fingerprints; the disadvantage of these studies is not focusing on improving image quality. Because they use features of singularity points position. This leads to a great influence on the accuracy of the system. The Galton-Henry classification scheme is used to classify fingerprints. This method is presented in Reference [20]. They use rotation-invariant distance, then match this distance between the FingerCode sample set with the new fingerprint pattern. During the classification process, the rotation-invariant distance extraction takes place in parallel with the training process. This gives the advantage of fast system time. The Random Forest (RF) algorithm is used to handle large amounts of data and multi-class problems in classification [21]. This algorithm is an ensemble of randomized decision trees that resolves a regression task [22] and a classification task [23–25] The most prominent application of random forest is the detection of human body parts from depth data [26]. This application demonstrates the practicability of random forests for machine learning problems in the real world.

There are many known ways to improve the performance of fingerprint classification systems. In this paper, we proposed a combination of computer vision algorithm and traditional machine learning methods to treat such problems, which we have successfully research in other fields [27–29]. Moreover, we also designed and implemented image processing methods with denoising and increasing edge using the Canny method.

3. Proposed Method

3.1. Fingerprint Image Pre-Processing and Enhancement

There have been many previous studies on solving the problem of a noisy or distorted image dataset. But, image pre-processing is always a challenging field for researchers to develop image quality enhancement algorithms. We focused on image pre-processing techniques based on computer vision techniques, such as image denoising to consider real-world images from an imperfect image-capturing device.

Traditionally, image-denoising methods are often used to treat noisy images [30]. Denoising suppresses the small details and perturbations and enhances the edges. This operation can basically

be represented as a blurring of the image, followed by the enhancement of the edges. Therefore, the resultant image emphasizes edges and suppressed details, thereby suppressing the noise in the image.

A dataset is directly collected with much noise, especially noise with an extremely high percentage of impulses, which brings a significant challenge for image denoising. So, in the noise filter step we propose a filter method based on the convolutional neural network (CNN) [31]. This process includes two main steps. First, we develop a pre-processing step for noisy images using non-local information. Then, the pre-processed images are divided into patches and used for CNN training, leading to a CNN denoising model for future noisy images, to detect the noisy pixels of the image and then smooth them using a Gaussian filter method. In the CNN training step, the pre-processed images are divided into overlapping patches. We use these patches as input for the convolutional neural network. Our network has three layers; in each layer, we define a set of filters and operators to generate mappings. The convolutional result of each patch is corresponding to a n-dimensional feature map. We define a convolutional layer to predict enhancing patches and reconstruct them as a result image in the third layer. In this work, the Gaussian algorithm [32] used to filter noise with mask 3×3, then a canny algorithm [33] is applied to improve enhance information edge. Therefore, when the morphology operation has processed all image, the result will be better. Image pre-processing steps are as follows (Figure 1).
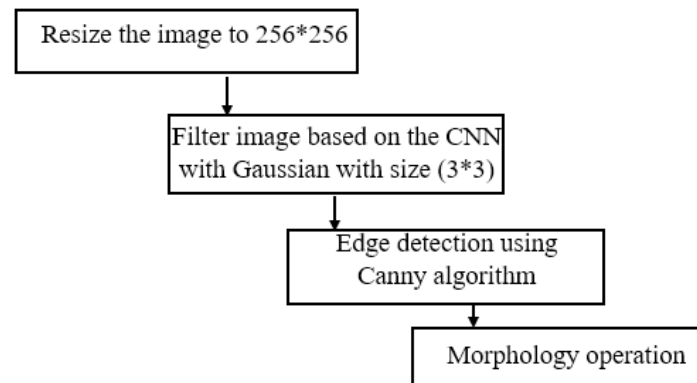

Figure 1. Block diagram of fingerprint image pre-processing.

The result of image pre-processing of a fingerprint shown in Figure 2. Figure 2a,b show the results of the equalization histogram and noise filter based on CNN, Figure 2c,d show the results of edge detection and morphology.
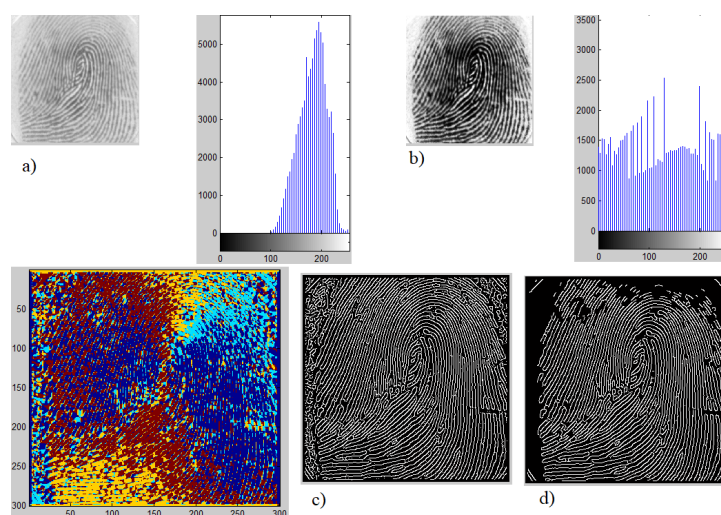

Figure 2. The result of image pre-processing of a fingerprint. (**a**,**b**) show the results of the equalization histogram and noise filter based on CNN, (**c**,**d**) show the results of edge detection and morphology

3.2. Types of Fingerprint and Features Extraction

In this work, we classify types of fingerprint into 3 classes—arch, loop and whorl.

- **Arch**: These occur in about 5% of the encountered fingerprints. The identifying features of this arch are that the fingerprints have overlapping shapes that form layers and have a mountain-like peak. Arch of fingerprints are divided into several categories—AS (the lines are stacked on top of each other, unconcerned, no intersection.), AE (a combination of whorl and arch group, the distance from the center to the intersection of eagles is less than 5 veins), AU, AR (the combination of the loop group with arch, the distance from the center to the intersection is less than 5 fringe lines) as shown in Figure 3.
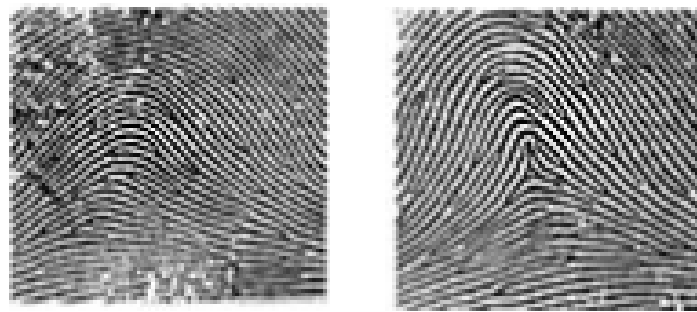

Figure 3. Example type of fingerprint is arch.

- **Loop**: It is called the loop (can be seen in almost 60% to 65% of fingerprints worldwide) fingerprint because it is shaped like a water wave with the following features—the ridges make a backward turn in loops, triangular with a center and an intersection. Divided into two types: *RL—Radial Loop*: Top of the triangle facing the pinky finger. It looks like a stream of water flowing downwards (on the little finger). This type accounts for about 6% of fingerprints worldwide. *UL—Ulnar Loop:* The top of the triangle faces the thumb. It is shaped like a stream of water flowing backward (thumb direction). This form only accounts for 2% of fingerprints worldwide. A loop pattern has only one delta as shown in Figure 4.
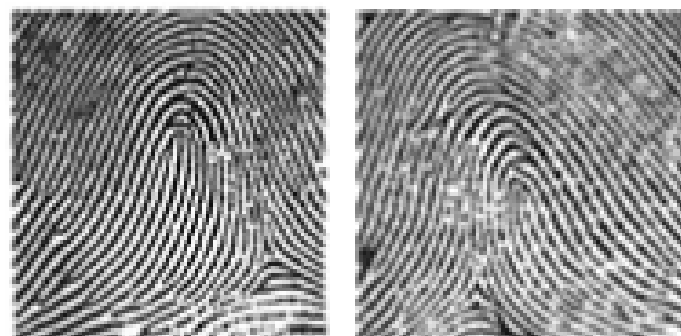

Figure 4. Example type of fingerprint is loop.

- **Whorl**: This fingerprint only accounts for about 25% to 35% of fingerprints worldwide. Whorl pattern identification is that they have one circuit and 2 Delta (intersection) as shown in Figure 5.
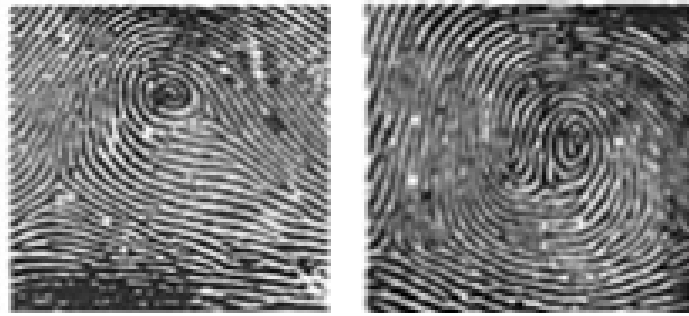
Figure 5. Example type of fingerprint is whorl.

Feature extraction of singularity. On the fingerprint there are areas with unusual structures compared to other areas. They often have a parallel structure called a singularity. There are two types of singularity—core and delta. To extract singularity characteristics, we proceed as follows:

- Step 1. Input image then resize image to 256×256
- Step 2. Fingerprint pre-processing and enhancement
- Step 3. At each pixel, the gradient is calculated in two directions x and y are G_x and G_y based on the Formula (1):

$$\varphi = \frac{1}{2} \tan^{-1} \left[\frac{\sum_{i=1}^{W} \sum_{j=1}^{W} 2G_x (i,j) G_y (i,j)}{\sum_{i=1}^{W} \sum_{j=1}^{W} \left(G_x^2 (i,j) - G_y^2 (i,j) \right)} \right] \tag{1}$$

- Step 4. Identify singularity points using the Pointcare index [15]. Pointcare index at the pixel with coordinates (i,j) is the sum of the deviations of direction of adjacent points, calculated as follows in Equation (2):

$$PC (i,j) = \sum_{k=0}^{N_p-1} \Delta (k) \tag{2}$$

$$\Delta (k) = \begin{cases} d(k); |d(k)| < \frac{\pi}{2} \\ d(k) + \pi; d(k) \leq -\frac{\pi}{2} \\ d(k) - \pi \end{cases} \quad d(k) = \varphi(x_{k+1}, y_{k+1}) - \varphi(x_k, y_k), \tag{3}$$

where φ is the gradient at pixels in two directions.

Based on the Pointcare index, we can identify singularity points as follows in Equation (4):

$$PC(i,j) = \begin{cases} 0^0; (i,j) \text{ is not singularity} \\ 360^0; (i,j) \text{ is whorl} \\ 180^0; (i,j) \text{ is loop} \\ -180^0; (i,j) \text{ is delta} \end{cases} \tag{4}$$

- Step 5. Save and create fingerprint features vector.

4. Classification Fingerprint Based on Random Forest and Decision Tree with Singularity Features

Fingerprint classification is a multi-class classification problem, the feature obtained by unsupervised learning and type of fingerprints (labels) is selected as the new input data for training a multi-classifier by way of a supervised method. In this work there are 3 labels (arch, loop and whorl). For fingerprint classification, a relatively small number of features extracted from fingerprint images. Here, we choose the orientation field base on the gradient and identify singularity features using the Pointcare index as our classification. The machine learning algorithms chosen to training module include Random Forest [34] and Support vector machine [35] (Figure 6).
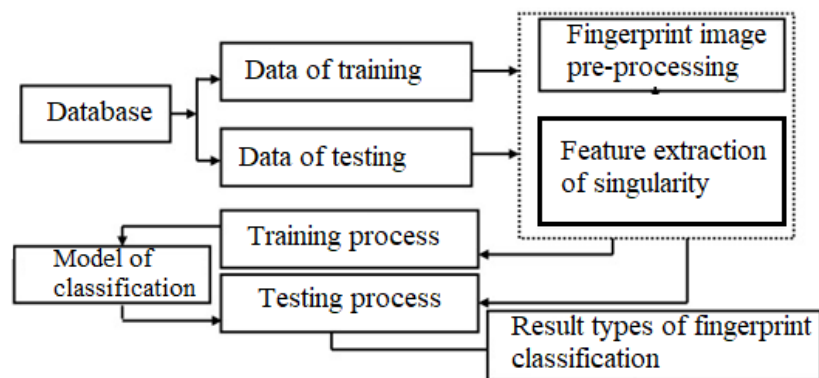
Figure 6. Block diagram of fingerprint classification using method proposed.

Parameters of Random Forest for Fingerprint Classification.

- The number of trees to train model = 2000.
- The function to measure the quality of a split - Gini Impurity.
- Bootstrap samples = True.

Parameters of Support Vector Machine for Fingerprint Classification.

- Penalty parameter to measure error term = 1.0.
- Kernel: basis functions.
- Shrinking heuristic = True.

5. Result of Experimentation

5.1. Database

Three databases were used in this paper which are—the Fingerprint Verification Competition (FVC) 2000, 2002 and 2004 [36–38]. The database consists of 788 learning images and 100 test images. Our database (HLG) has 500 fingerprint images taken from 100 different fingers; an image of each finger was taken 5 times. The resolution of each image is 300 dpi and the size is 256×256. From this number of images, we divide them to create training data of 400 images and 100 images for matching and database 3, the NIST-DB4 database, is used for testing classification accuracy by most of the algorithms, which consists of 4000 fingerprint images (image size is 512×512 (Figure 7). All experiments were carried out using a Win10 PC with Intel Core i3 CPU @ 3.00 GHz and 2.00 GB RAM.

Figure 7. Examples of fingerprint in database.

5.2. Analysis Results of Experimentation

In this article, factors (Accuracy, Precision, Recall) are used to evaluate the performance of a fingerprint classification system using machine learning methods (Random Forest algorithm, Support Vector Machine).

- Accuracy is a system to measure the degree of closeness of measurements of a quantity to that quantity's actual (true) value.

$$Acc = (True positive + True Negative)/(Total number of elements)$$

- Precision is the fraction of retrieved documents that are relevant to the find:

$$Precision = (True Positive)/(True positive + False Positive)$$

- Recall in information retrieval is a fraction of the documents that are relevant to a query that is successfully retrieved.

$$Recall = (True Positive)/(True Positive + False Negative)$$

The results shown in Figures 8–11 demonstrated that the algorithms work stably and with high accuracy. For example, NIST-DB4 consists of a large number of images (>4000), therefore the algorithms are provided with a wide range of input data, which allows them to study more and achieve an accuracy of >97%.
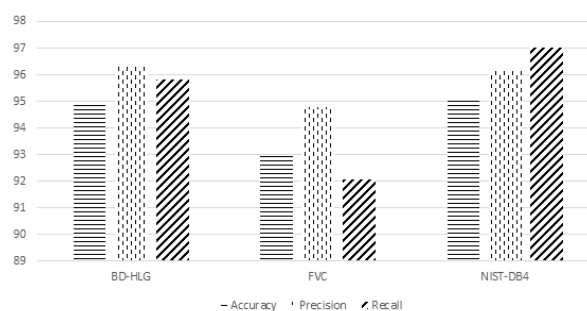

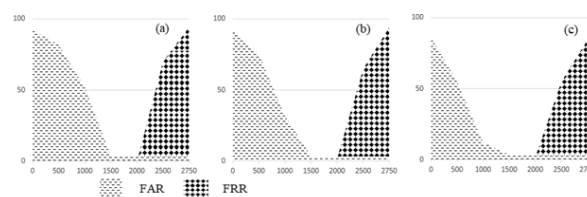**Figure 8.** Result of classification using Support Vector Machine algorithm of 3 databases.

Figure 9. Analysis rate errors of Support Vector Machine algorithm in 3 databases ((**a**) BG-HLG, (**b**) FVC, (**c**) NIST-BD4).
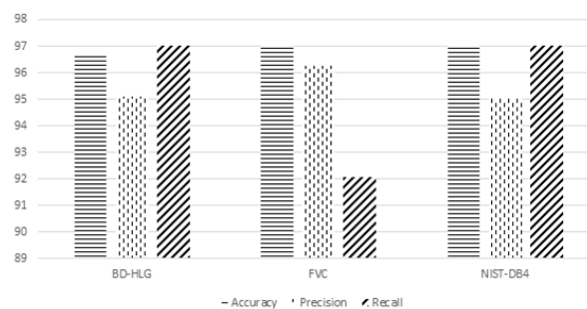

Figure 10. Result of classification using Random Forest algorithm of 3 databases.
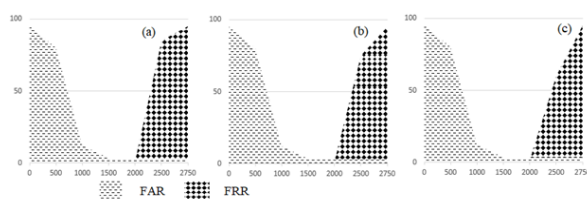

Figure 11. Analysis rate errors of Random Forest algorithm in 3 databases ((**a**) BG-HLG, (**b**) FVC, (**c**) NIST-BD4).

To demonstrate the robust accessibility of the two proposed machine learning algorithms—(Random Forest (RF) and Support Vector Machine (SVM)—we have experimented with 3 data sets with 2 models of CNN and k-NN.

The CNN model is trained with a learning rate of 0.01 using a softmax activation function in pooling class. The Relu trigger function is used for all the hidden layers of the convolutional layer, the most learning times of the network is 100. The basic network architecture consists of—Input layer–Layers [Convolution layer–MaxPooling class–Activation class]–Output layer. When training the network, we chose parameter Batch size = 32 to select the sample size to be learned and to adjust the parameters in the network layers to have the best accuracy. The maximum number of network training sessions is 100. The model with the best classification results on the training set will be selected and tested on test sets for evaluation.

We use the kNN model to experiment with many k parameters (k = 1, 3, 5, 7) to choose the best results. In the model aggregation method, we ran experiments with different numbers of decision trees to select the best results. The number of decision trees tested was from 150 to 300 trees. Through experiments, we can see that the RF algorithm for building 200 decision trees will ensure high accuracy. The SVM algorithm uses a linear multiplier function that has the best results for MGE data sets. The results are obtained from classification models (with optimal parameters) as shown in Tables 1–3 and ROC analysis Figure 12. The tables show the effect of increasing the number of trees in the ensemble. For both, increasing trees requires more time to learn but also provide better results in terms of Mean Squared Error (MSE) is calculated as follows in Equation (5):

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (f(x_i) - y_i)^2, \tag{5}$$

where n is the number of test examples, $f(x_i)$ the classifier's probabilistic output on x_i and y_i are actual labels.

Table 1. Results of classification algorithms: RF, SVM, CNN, k-NN of BD-HLG database.

	RF	SVM	CNN	k-NN
Training time (s)	8H15M	7H40M	8H32M	5H28M
Correct rate (%)	96.25	95.76	95.82	91.90
MSE	0.274	0.321	0.303	0.481

Table 2. Results of classification algorithms: RF, SVM, CNN, k-NN of FVC database.

	RF	SVM	CNN	k-NN
Training time (s)	12H30M	9H06M	14H05M	7H50M
Correct rate (%)	97.78	95.83	96.11	92.05
MSE	0.215	0.304	0.267	0.668

Table 3. Results of classification algorithms: RF, SVM, CNN, k-NN of NIST-DB4 database.

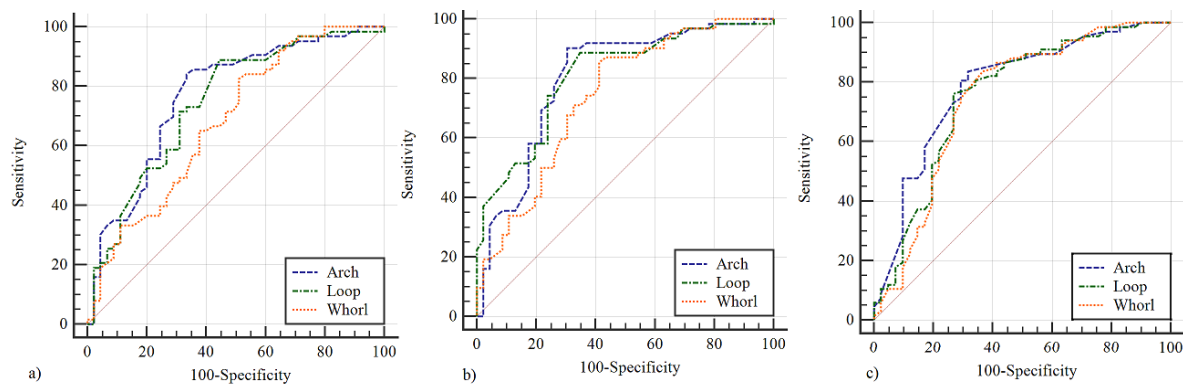|	RF	SVM	CNN	k-NN
Training time (s)	18H47M	15H40M	20H31M	10H38M
Correct rate (%)	97.86	96.81	96.90	92.88
MSE	0.255	0.317	0.240	0.719

Figure 12. Analysis ROC of the CNN (**a**), Support Vector Machine (**b**), Random Forest (**c**).

The results of the classification of 3 databases with the criteria of Precision, Recall, Accuracy evaluation of Random Forest and SVM algorithms show that the Random Forest algorithm has the best accuracy (>96.75%) for all 3 criteria. The SVM algorithm and CNN models have the best accuracy (>95.5%) for 2/3 of the database. The K-NN model has the lowest accuracy (<93%). The RF algorithm and CNN models need a lot of training time due to the process of tree construction and the interpretation of many cases.

6. Conclusions

A fingerprint classification method is proposed in this paper, based on two machine learning algorithms—Random Forest and Support Vector Machine. Both algorithms demonstrate the advantage of machine learning in the classification of objects (the accuracy ≥96%). Computer vision techniques are used in combination to improve image pre-processing methods with CNN technology. This helps to increase the quality of input images for the processing system. This study used the singularity feature of fingerprints. For future plans we will be developing algorithms to extract more features and we will be using deep learning methods to optimize the system.

References

1. Alpaydin, E. *Introduction to Machine Learning*; The MIT Press: London, UK, 2009; p. 584.
2. Bertrand, C.; Ernest, F.; Zhang, H.H. *Principles and Theory for Data Mining and Machine Learning*; Springer Press: New York, NY, USA, 2009; p. 786, ISBN 978-0-387-98135-2.
3. Bishop, C. *Pattern Recognition and Machine Learning*; Springer Press: New York, NY, USA, 2006; p. 738
4. Kotsiantis, S.B. Supervised machine learning: A review of classification techniques. *MathSciNet* **2007**, *31*, 249–268.
5. Bruijne, M.D. Machine learning approaches in medical image analysis: From detection to diagnosis. *JACR Med. Image Anal.* **2016**, *33*, 94–97. [CrossRef] [PubMed]
6. Boykov, Y. Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images. In Proceedings of the Eighth IEEE International Conference on Computer Vision, Vancouver, BC, Canada, 7–14 July 2001; pp. 105–112.
7. Tomin, N.V.; Kurbatsky, V.G.; Sidorov, D.N.; Zhukov, A.V. Machine Learning Techniques for Power System Security Assessment. *IFAC-PapersOnLine* **2016**, *49*, 445–450, doi:10.1016/j.ifacol.2016.10.773. [CrossRef]

8. Milan, S.; Vaclav, H.; Roger, B. Image pre-processing. *Image Processing, Analysis and Machine Vision*; Springer: Boston, MA, USA, 1993; pp. 56–111.

9. Strobl, C. An Introduction to Recursive Partitioning: Rationale, Application and Characteristics of Classification and Regression Trees, Bagging and Random Forests. *Psychol. Methods* **2009**, *14*, 323–348. [CrossRef] [PubMed]

10. Cristianini, N.; Shawe-Taylor, J. *An Introduction to Support Vector Machines (and Other Kernel-Based Learning Methods)*; Cambridge University Press: Cambridge, UK, 2000; p. 190.

11. Zhukov, A.V.; Sidorov, D.N.; Foley, A.M. Random Forest Based Approach for Concept Drift Handling. In *Analysis of Images, Social Networks and Texts*; Springer: Berlin, Germany, 2016; p. 661.

12. Maio, D.; Maltoni, D. A structural approach to fingerprint classification, Pattern Recognition. In Proceedings of the 13th International Conference on Pattern Recognition, Vienna, Austria, 25–29 August 1996; Volume 3, pp. 578–585.

13. Mohamed, S.M.; Nyongesa, H.O. Automatic fingerprint classification system using fuzzy neural techniques, Fuzzy Systems. In Proceedings of the 2002 IEEE World Congress on Computational Intelligence, 2002 IEEE International Conference on Fuzzy Systems, Honolulu, HI, USA, 12–17 May 2002; Volume 1, pp. 358–362.

14. Zhang, Q.; Yan, H. Fingerprint classification based on extraction and analysis of singularities and pseudo ridges. *Pattern Recognit.* **2004**, *37*, 2233–2243. [CrossRef]

15. Hong, J.H.; Min, J.K.; Cho, U.K.; Fingerprint classification using one-vs-all support vector machines dynamically ordered with Bayes classifiers. *Pattern Recognit.* **2008**, *41*, 662–671. [CrossRef]

16. Senior, A. A combination fingerprint classifier. *IEEE Trans. Pattern Anal. Mach. Intell.* **2001**, *23*, 1165–1174. [CrossRef]

17. Karu, K. *A Jain Fingerprint Classification Pattern Recognition*; Elsevier: Amsterdam, The Netherlands, 1996; pp. 389–404.

18. Nyongesa, H.; Al-khayatt, S. Fast robust fingerprint feature extraction and classification. *J. Intell. Robot. Syst.* **2004**, *40*, 103–112. [CrossRef]

19. Nagaty, K. Fingerprints classification using artificial neural networks: A combined structural and statistical approach. *Neural Networks* **2001**, *14*, 1293–1305 [CrossRef]

20. Guo, T.; Liu, X.; Shao, G. Fingerprint Classification Based on Sparse Representation Using Rotation-Invariant Features. In *Proceedings of the 2012 International Conference on Information Technology and Software Engineering*; Springer: Berlin/Heidelberg, Germnay, 2013; pp. 631–641.

21. Schroff, F.; Criminisi, A.; Zisserman, A. Object class segmentation using randomforests. In Proceedings of the British Machine Vision Conference 2008, Leeds, UK, 1–4 September 2008; pp. 1–10.

22. Everingham, M.; Van Gool, L.; Williams, C. The pascal visual object classes (voc) challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [CrossRef]

23. Leistner, C.; Saffari, A.; Santner, J.; Bischof, H. Semi-supervised random forests. In Proceedings of the 2009 IEEE 12th International Conference on Computer Vision, Kyoto, Japan, 29 September–2 October 2009; Volume 26, pp. 506–513.

24. Lepetit, V.; Lagger, P.; Fua, P. Randomized trees for real-time keypoint recog-nition. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA, 20–25 June 2005; Volume 27, pp. 775–781.

25. Lowe, D. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **2004**, 91–110. [CrossRef]

26. Schulter, S.; Leistner, C.; Roth, P.; Bischof, H.; Van Gool, L. On-line hough forests. In Proceedings of the British Machine Vision Conference (BMVC 2011), London, UK, 29 August–2 September 2011.

27. Nguyen, T.H.; Nguyen, T.L.; Dreglea, A.I. Machine learning algorithms application to road defects classification. *Intell. Decis. Technol.* **2018**, *12*, 59–66. [CrossRef]

28. Nguyen, T.H.; Nguyen, T.L.; Dreglea, A.I. Robust approach to detection of bubbles based on images analysis. *Int. J. Artif. Intell.* **2018**, *16*, 167–177.

29. Nguyen, T.H.; Nguyen, T.L. ROC curve analysis for classification of road defects. *BRAIN Broad Res. Artif. Intell. Neurosci.* **2019**, *10*, 65–73.

30. Derin, H. Modeling and segmentation of noisy and textured images using Gibbs random fields. *IEEE Trans. Pattern Anal. Mach. Intell.* **1987**, *9*, 39–55. [CrossRef] [PubMed]

31. Felix, J.X.; Vishnu, N.B.; Marios, S. Local binary convolutional neural networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; Volume 1, pp. 473–484.

32. Haddad, R.A. A Class of Fast Gaussian Binomial Filters for Speech and Image Processing. *IEEE Trans. Acoust. Speech Signal Process.* **1991**, *39*, 723–727. [CrossRef]

33. Canny, J. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **1986**, *8*, 679–698. [CrossRef] [PubMed]

34. Breiman, L. Random forests. *Mach. Learn.* **2001**, *1*, 5–32. [CrossRef]

35. Kim, K.J. Financial time series forecasting using support vector machines. *Neurocomputing* **2003**, *55*, 307–319. [CrossRef]

36. Fingerprint Verification Competition, "FVC 2000". Available online: http://bias.csr.unibo.it/fvc2000/ (accessed on 1 August 2000).

37. Fingerprint Verification Competition, "FVC 2002". Available online: http://bias.csr.unibo.it/fvc2002/ (accessed on 14 August 2002).

38. Fingerprint Verification Competition, "FVC 2004". Available online: http://bias.csr.unibo.it/fvc2004/ (accessed on 1 August 2004).