

## Article

# Classification of Japanese Handwritten Characters Using Biometrics Approach

Piotr Szymkowski <sup>1,\*</sup>, Khalid Saeed <sup>1,2,†</sup>, Łukasz Szymkowski <sup>1,†</sup> and Nobuyuki Nishiuchi <sup>3,†</sup>

<sup>1</sup> Faculty of Computer Science Białystok, University of Technology, ul. Wiejska 45 A, 15-351 Białystok, Poland; k.saeed@pb.edu.pl (K.S.); lukaszszymkowski@gmail.com (Ł.S.)

<sup>2</sup> Department of Electronics and Computation Sciences, Universidad de la Costa, Barranquilla 080002, Colombia

<sup>3</sup> Faculty of Systems Design, Tokyo Metropolitan University, 6-6 Asahigaoka, Hino 191-0065, Japan; nnishiuc@tmu.ac.jp

\* Correspondence: prszymkowski@gmail.com

† These authors contributed equally to this work.

**Abstract:** The following paper presents a solution to the problem of offline recognition of Japanese characters. Minutiae and other features extractable from handwriting images have been used to recognize individual characters. The solution presented by the authors uses minutiae to recognise single Japanese characters. Due to the complexity of this typeface, the solution presented can be used to recognise archaic characters, from old documents or also works of art. Neural Networks and hybrid classifiers based on five basic types of classifiers, i.e., k-nearest neighbour method, decision trees, support vector machine, logistic regression and Gaussian Naive Bayes classifier have been developed for classification. The study was conducted on Hiragana, Katakana and Kanji characters (ETL9G Database). The accuracy value obtained was 99.934%. The authors present what is probably the first algorithm using minutiae to recognize Japanese handwriting.

**Keywords:** image processing; text recognition; japanese handwriting



**Citation:** Szymkowski, P.; Saeed, K.; Szymkowski, Ł.; Nishiuchi, N. Classification of Japanese Handwritten Characters Using Biometrics Approach. *Appl. Sci.* **2024**, *14*, 225. <https://doi.org/10.3390/app14010225>

Academic Editors: Waleed Abdulla and Iman Ardekani

Received: 19 November 2023

Revised: 23 December 2023

Accepted: 24 December 2023

Published: 26 December 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Japanese, Chinese and Korean are among the leading languages in Asia. The way of writing in these countries differs significantly from the letters used in the Latin alphabet. The main difference is the complexity of the characters used. The number of lines and unusual shapes of characters may result in difficulties during recognition attempts. An additional complicating aspect is the number of characters. There are three methods of notation used in Japanese [1]. Two of them are syllabaries, that is, one character means one syllable. There are 46 characters in each syllabary. The third type is Kanji characters. One written character has one meaning. For everyday use, knowledge of about 2000 characters is required.

With such a large number of complex characters, it seems that methods similar to fingerprint biometric recognition will be possible. The research will use minutiae, which are feature points typically used in biometric recognition [2,3]. The proposed solution solves the offline optical character recognition (OCR) problem. It means that as input to the algorithm we receive only the image of the character. There is no possibility of obtaining features characteristic of online methods, i.e., direction and order of lines or the force of pen pressure on paper. Despite the impossibility of using the previously mentioned features, it is possible to use other ones, which may give a chance to obtain satisfactory results. The presented solution can be used in the future to recognize archaic characters from old books, documents, or paintings.

### 1.1. State of the Art

The methods most commonly used to recognize Japanese handwriting will be presented below.

#### 1.1.1. Semi Markov Conditional Random Fields

The first algorithm presented is based on semi-Markov conditional random fields [4]. With its help, tests were performed on databases: CASIA-OLHWDB with Chinese characters, and TUAT Kondate with Japanese characters. Accurate rates were 94.54% and 94.55%. One-step algorithms based on hidden Markov models use the division of lines of text into segments of the same size. These segments are then described and classified. This type of method does not obtain accurate results about the shapes of characters, so the classification itself does not provide satisfactory results. The two-stage approach besides similar partitioning using hidden Markov models, implies the use of a different classifier to recognize segmented text [5,6]. The presented solution consists of dividing lines of text into smaller fragments. These fragments can then be combined into sub-paths. The maximum length of a sub-path is determined by the variable  $m$ , called clique [7]. For the classification itself, the authors use three feature functions: character classification, geometric context and linguistic context. This solution relies heavily on the support of Markov chains. This type of solution, unlike the one presented by the authors in this publication, may have difficulty recognising texts in which the sequence of characters is random, but contains uncommon combinations of characters, such as in names or surnames.

#### 1.1.2. Lexicon-Driven Segmentation and Recognition

This work solves the problem of postal address recognition. The address is recognised as a whole since there are no gaps in it [8]. The lexicon used by the authors contains more than 111,000 phrases. Based on the stored phrases, segmentation of individual characters is performed. Finally, using the knowledge about possible combinations of characters occurring after each other, the solution giving the highest chance of correctness is chosen. During the research on more than 3500 images of addresses, the method proposed by the authors allowed to achieve the correctness of address recognition in 83.68% of cases. The algorithm uses morphological filters to smooth the edges as well as narrow the characters. Unlike the authors' algorithm, the skeletonisation algorithm is not used, and thus the solution presented in the publication does not take full advantage of the sealed image.

#### 1.1.3. Deep Learning Approach

Currently, deep learning algorithms, in particular convolutional neural networks, are used when attempting handwriting recognition. Their operation is based on the representation of human senses. When using this algorithm, the user is relieved to create feature vectors. Convolutional neural network by way of convolution, or splicing operation. Successive layers of the neural network are designed to generate a set of features without user intervention. Subsequent layers of the convolutional network create some generalisation of the information from the previous layer. In the case of the publication created by Xu-Yao Zhang, Yoshua Bengio, and Cheng-Lin Liu [9], the previously mentioned convolutional network was applied to Chinese handwriting recognition. An additional aspect considered was the acquisition of a Direction Decomposed Feature Map. The authors report an accuracy of 97.37%. The presented solution works on images in shades of gray, while the algorithm presented by the authors can work on data also containing information in all colours.

## 2. Used Image Processing Algorithms and Methods

Image processing algorithms have been used to prepare the samples to create feature vectors from them. This chapter will present the most important of the methods used.

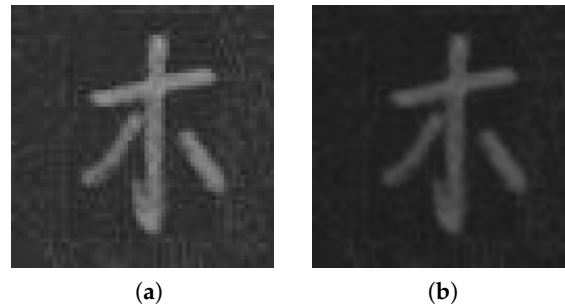
### 2.1. Gabor Filter

The Gabor filter is a linear filter used for texture analysis [10,11]. It allows the image to be filtered while maintaining a selected frequency range (Figure 1). This filter can operate in both frequency domain and spatial domain [12–14]. In the spatial domain, its formula is as follows:

$$g(x, y) = s(x, y) * w_r(x, y) \quad (1)$$

where

- $s(x, y)$ —carrier sine wave,
- $w_r(x, y)$ —two-dimensional Gaussian function, envelope.



**Figure 1.** Original sample (a), sample after gabor filter application (b).

### 2.2. Bilateral Filter

The bilateral filter is a non-linear filter. It smoothens and blurs the image while preserving edges [10,11]. The brightness of the considered (central) pixel under a structural object is calculated by applying a weighted average of the brightnesses of pixels located in the vicinity of the considered pixel [15]. The weights depend on the distance and the difference in brightness between the considered pixel and particular pixels of its surroundings (Figure 2). The filtering procedure is a convolution of the function with the mask function:

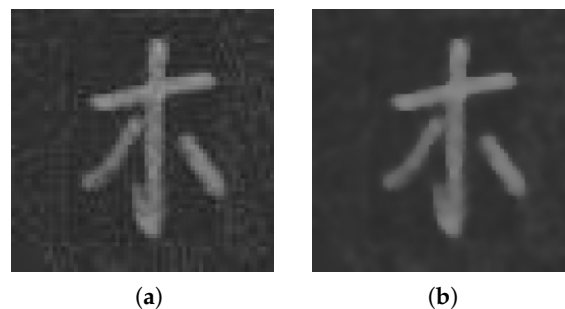
$$f(x_0) = \frac{1}{k} \sum_{i \in R} f(x_i) * h_D(x_0 - x_i) * h_I(x_0 - x_i) \quad (2)$$

and the normalising factor function looks like this:

$$k = \sum_{i \in R} h_D(x_0 - x_i) * h_I(x_0 - x_i) \quad (3)$$

where

- $h_D$ —distance function,
- $h_I$ —intensity function,
- $R$ —surroundings of the point  $x_0$ ,
- $x_0$ —point under consideration.



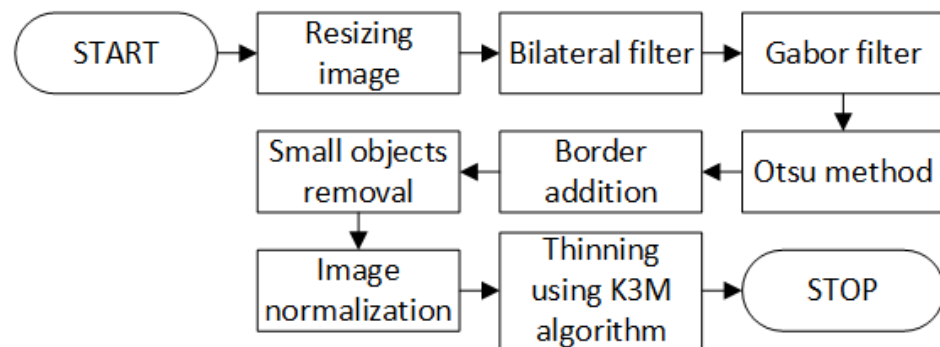
**Figure 2.** Original sample (a), image after bilateral filter application (b).

### 3. Authors' Solution

The authors' algorithm is divided into three stages: preprocessing, feature vector generation, and classification. Each of these stages consists of multiple algorithms and methods.

#### 3.1. Preprocessing

The most important step when working with image data is preprocessing (Figure 3). The purpose of preprocessing is to prepare samples for further processing steps or feature extraction. The samples used by the authors during their research come from the ETL character database [16].



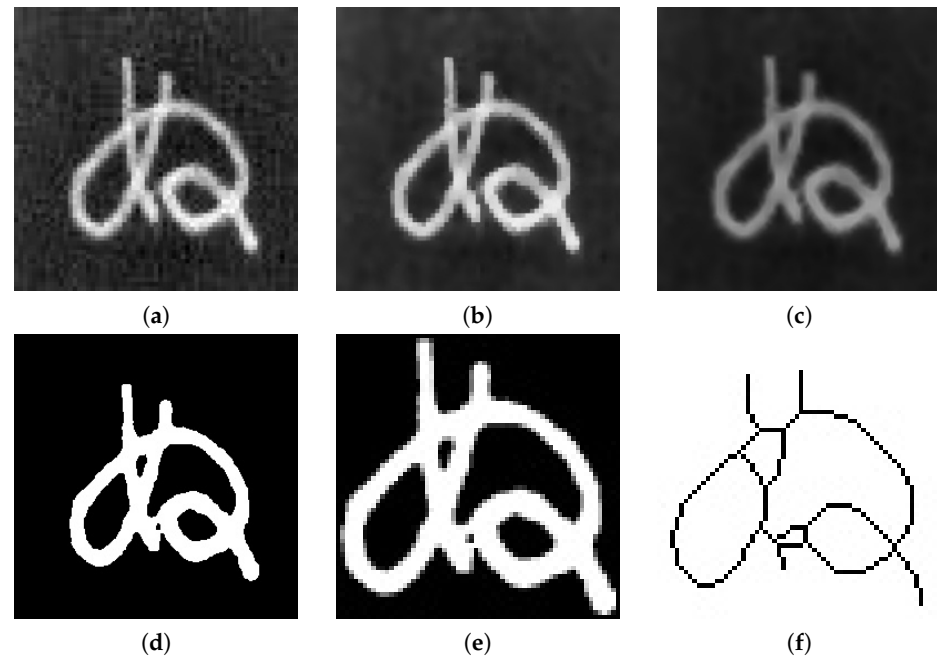
**Figure 3.** Diagram of preprocessing process.

Each sample has the same size at the beginning ( $62 \times 62$  pixels). The first step is to enlarge the images to facilitate further algorithms. The images are resized to  $256 \times 256$  pixels. The next step is to apply a bilateral filter. In this case, the bilateral filter mask is 9 pixels long. The parameters for the bilateral filter were selected on the basis of trials conducted by the authors and the chosen parameters are: 15 -diameter of the pixel neighbourhood, sigmaColor:75 and sigmaSpace:75. After applying this filter, the initial noise appearing both in the background and the objects themselves (the signs under consideration) is removed. The next step is to apply the Gabor filter (ksize = 35, sigma = 3.0, lambda = 10.0, gamma = 0.5), which makes the letter more visible in the image and then Otsu binarisation is applied.

After applying these two algorithms, we obtain a binary image of Japanese characters. At this stage, we can already see the distinction between the object (white colour) and the background (black colour). For further operations, a frame of one-pixel width is added to the images. The next step is to remove small objects [17]. This is the step that allows for the final removal of noise. It is conducted by calculating the object size of the image. If the mentioned size is bigger than the threshold value then the object is removed. In the case of the presented research, the threshold value is 20. Next, normalisation is performed. Extreme rows and columns that do not carry any information are removed from the image. The image is resized to  $62 \times 62$  pixels and a frame of one pixel width is added again. The final image has a height and width of 64 pixels. The last operation is to apply the K3M-modified thinning algorithm. After its application, an image representing the skeleton of the character is obtained. The width of each line in the image is 1 pixel (Figure 4) [18].

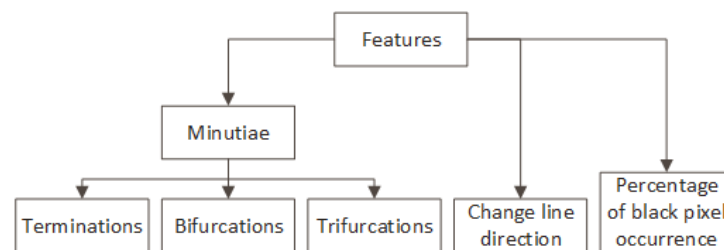
#### 3.2. Feature Vector Creation

Based on the images obtained in the previous step, feature vectors are created. Most of the features used are minutiae, which are feature points used in biometric recognition using fingerprints. Types of minutiae used in this work are beginnings and ends, bifurcations, and trifurcations (Figure 5). Those minutiae are extracted by using a  $3 \times 3$  pixel mask. Each location in the mask is assigned a power of two. After applying the filter to the image, the sums obtained are checked against those corresponding to particular minutiae (Figure 6).

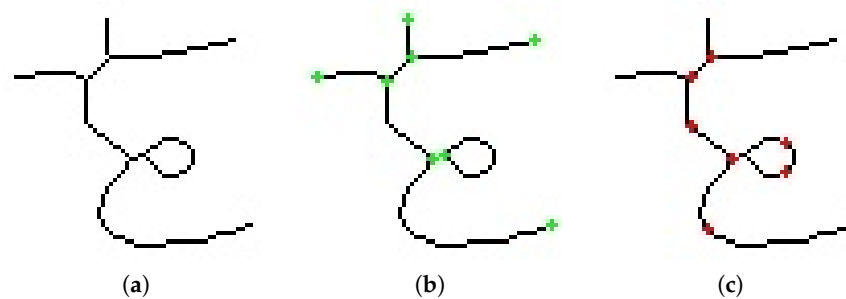


**Figure 4.** Original image (a), sample after bilateral filter application (b), image after gabor filter application (c), image binarised (d), sample normalised (e), image after K3M thinning algorithm application (f).

Other features considered are the number of object pixels present in the image and the change in the line direction. If the character is more complex then the number of dark pixels in the image increases.

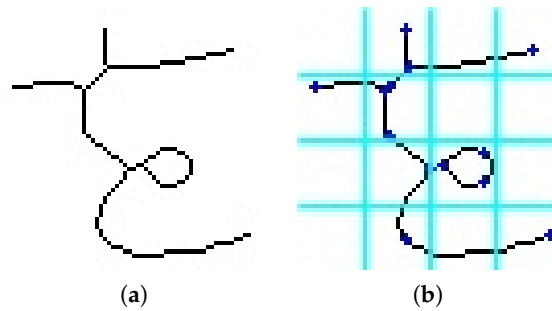


**Figure 5.** Features types.



**Figure 6.** Thinned image (a), minutiae found (b), change direction of the line found in image (c).

To obtain more information about the characters the image is divided into 16 segments (4 rows and 4 columns) and then the minutiae and the obtained features are divided according to their location (Figure 7). Finally, each vector has 86 values. The first 80 contain information about the number of each specific characteristic point occurrence. The remaining five contain information about the sum of all occurrences of each of the minutiae in the image. The last feature contains information about the class.



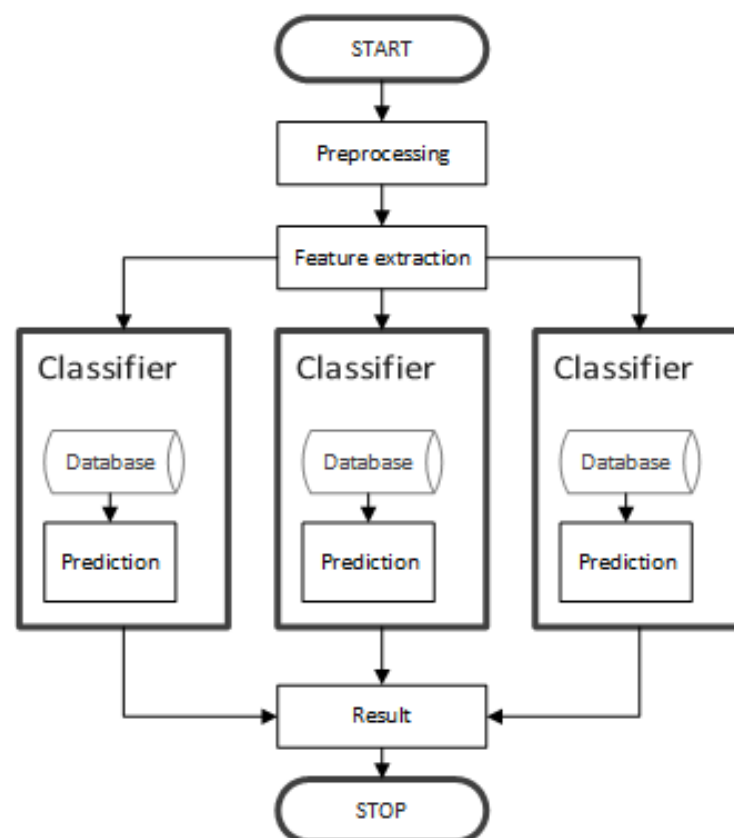
**Figure 7.** Thinned image (a), classification areas of features (b).

### 3.3. Classification

Two databases were used during the classification of handwritten Japanese characters. The first one contained over 32,000 Hiragana and Katakana characters collected from the ETL database. In the second, the entire contents of the ETL9G database (606,900 samples in 3036 classes) were applied to increase the number of processed images. Features vectors were normalised before classifiers were used. For the classification of the first database, the following classifiers were used:

- classifier based on decision trees,
- k-nearest neighbour method,
- classifier based on logistic regression,
- support vector machine (SVM),
- Gaussian Naive Bayes classifier.

Two multi-modal classifiers were created during the execution of the study: 3-modal and 5-modal. They work by obtaining feedback from the classifiers they contain and then a class is selected by voting. If there is a tie, a class is selected based on the classifier with the higher degree of accuracy (Figure 8).



**Figure 8.** Diagram for multimodal classifier solution.

The parameters of the classifiers were selected using the gridsearch method. The values of the selected parameters are shown in the table (Table 1). For the second database, the same classification algorithms were used as for the first database, but a sequential neural network was used instead of hybrid classification due to the low performance achieved. This database contains 606,900 samples (Hiragana, Katakana and Kanji), divided into two groups: training (404,600) and testing (202,300). This database contains 3036 decision classes. The created network has one input layer (85 neurons with relu activation function), three hidden layers (380, 759 and 1518 neurons with relu activation function) and one output layer (3036 neurons with softmax activation function).

**Table 1.** Classifiers comparison.

Classifier	Accuracy
Decision tree (max_depth = 15)	77.630%
k-NN (k = 11)	85.712%
SVM (kernel = rbf)	92.517%
Logistical Regression (solver = 'liblinear', max_iter = 150)	89.834%
Gaussian Naive Bayes classifier (smoothing = $1 \times 10^{-9}$ )	58.783%

#### 4. Results

During the study, the data used were divided into two groups: training and test in a ratio of 2:1. In a smaller database that means the training group contains over 20,000 feature vectors, and the test group contains over 10,000 feature vectors, which contain Hiragana and Katakana related data only. The first step was to check the classification results using standard classifiers (Table 1).

Among the classification algorithms used, the highest accuracy was obtained using the Support Vector Machine, which was 92.517%. The lowest was obtained using the Gaussian Naive Bayes classifier (58.783%).

The next step was to create two multimodal classifiers. The first one was based on all five classifiers mentioned in the earlier experiment. The second one used only three classifiers: the Support Vector Machine, the Decision Tree Classifier and the Gaussian Naive Bayes Classifier. These classifiers were selected on the basis of the research conducted. This composition resulted in the highest combined score.

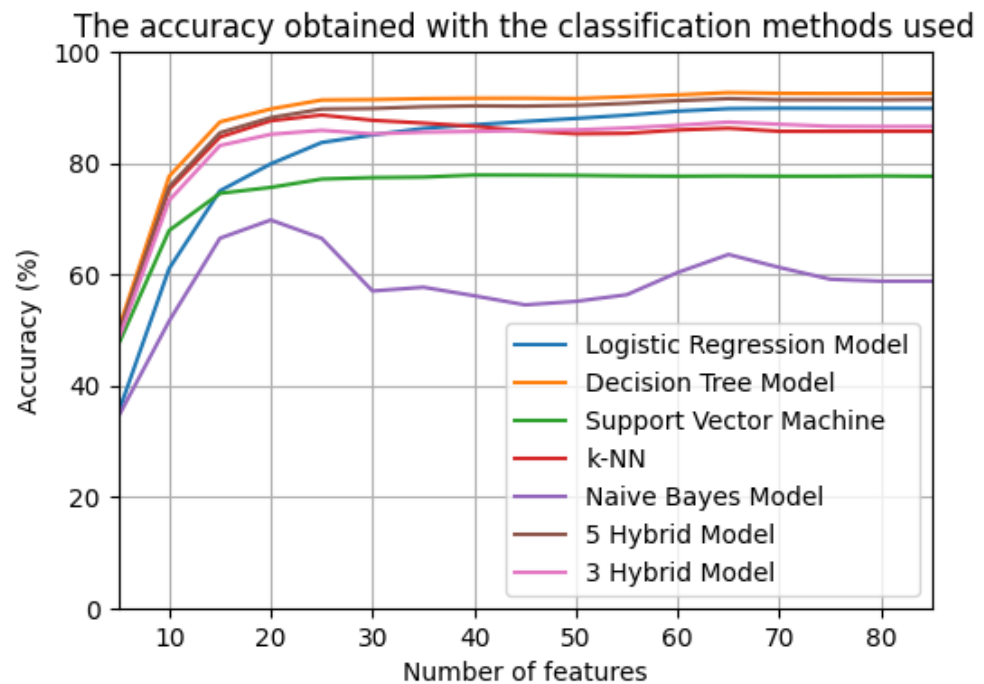
The obtained results for the five-modal classifier are higher than the three-modal classifier (Table 2). Results obtained with both trimodal and five-modal classifier are lower than those achieved by using Support Vector Machine. For this reason, it can be assumed that the use of multi-modular classifiers to classify characters using the authors' methods is unfounded. In the case of a tie for multimodal classifiers, the score taken into account is that obtained by the classifier with the highest level of accuracy.

**Table 2.** Classifiers comparison.

Classifier	Accuracy
Trimodal classifier	86.574%
Five-modal classifier	91.435%

Additionally, it was tested whether it is possible to reduce the number of parameters without affecting the quality of classification (Figure 9). From the graph shown, it is possible to reduce the number of parameters to 65 without losing accuracy in the results.





**Figure 9.** Accuracy achieved by different classification methods.

For the second database, the results obtained using standard classifications are much lower. The obtained results did not exceed 62% accuracy (Table 3). The authors conducted a study to select the best parameters. The parameters presented were experimentally selected.

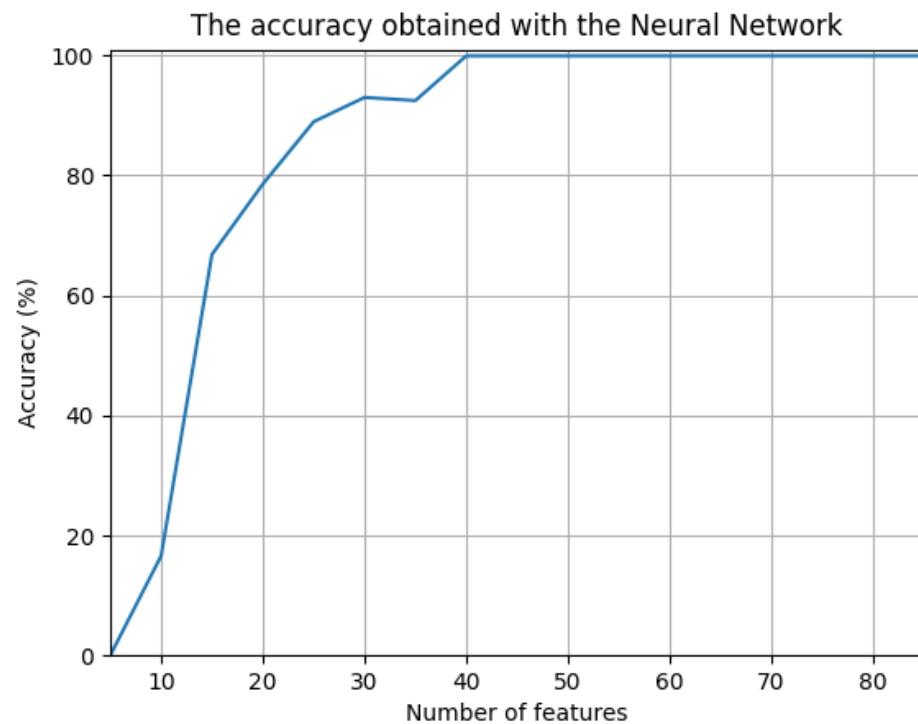
**Table 3.** Classifiers comparison.

Classifier	Accuracy
Decision tree (max_depth = 15)	19.836%
k-NN (k = 6)	33.059%
SVM (kernel = rbf)	43.234%
Logistical Regression (solver = 'liblinear', max_iter = 150)	61.399%
Gaussian Naive Bayes classifier (smoothing = $1 \times 10^{-9}$ )	58.783%
Neural Network	99.934%

A sequential neural network consisting of one input layer, three hidden layers and one output layer was used to obtain better results. In this way, a result of 99.934% accuracy was obtained. In addition, it was checked whether reducing the number of parameters in the feature vector would somehow affect the obtained result (Figure 10). In this case, optimal results can already be obtained when limiting the number of features to 40.

The result presented is one of the highest among the compared algorithms (Table 4). The last column shows the obtained result declared in the mentioned works. Comparing results between authors is difficult due to the use of different databases, but the closest comparison seems to be with authors using the ETL9B database. In this database, the images are already pre-segmented and binarised. The work presented by the authors shows one of the highest results achieved for the ETL9 database and this result is 99.934%.





**Figure 10.** Accuracy achieved by different classification methods.

**Table 4.** Solution comparison.

Authors	Approach	Database	Declared Achieved Score
Author's Solution	Minutia based	ETL9G	99.934%
N. Kato, M. Suzuki, S. Omachi, et al. [19]	Directional element feature and asymmetric Mahalanobis distance	ETL9B	99.42%
M.Mori, T. Wakahara, K. Ogura [20]	Extended Peripheral Direction Contrubutivity	ETL9B	95%
S. Tsuruoka, M. Hattori, M. F. b. A. Kadir, et al. [21]	Modified quadratic discriminate function and users dictionaries	ETL9B	82.5%
T-F. Gao, C-L. Liu [22]	LDA-Based Compound Distance	ETL9B	99.31%

## 5. Conclusions

The main achievement in the presented work is the adaptation of the biometric algorithm used in the recognition of fingerprints in a new environment. The results obtained make it possible to suspect that there is a possibility to apply it in the recognition of handwriting in old handwritten documents. The algorithm here only uses knowledge about the shape of the characters, it does not use knowledge about the context of the text and is therefore not limited to texts that make sense. In addition, results are presented using different classifiers, allowing a more accurate evaluation of the results. Due to the specificity of the language, as well as the developed algorithm, the presented solution should perform better in the recognition of complex writing, requiring a larger number of lines to be written. The part of the algorithm related to the database and its appearance was preprocessing, during which noise appearing in the images is removed. The images were sourced from the ETL9G database. For most other authors, the database used is ETL9B, which has black-and-white images, which no longer require cleaning.

Based on the obtained results, it can be concluded that recognition of Japanese characters using the proposed approach is possible. Using selected classification algorithms for

this purpose results in a high recognition rate, and combining selected classifiers can result in an increased recognition rate. The recognition accuracy (99.934%) is high despite the large number of classes (3036). The authors' next step will be to increase the size of the data used from different databases and sources. This will allow for a preprocessing algorithm that does not depend on database-specific appearance. The authors will also attempt other classification methods such as boosting or bagging.

**Author Contributions:** P.S. and Ł.S. wrote the main manuscript text and prepared figures. They conducted research on handwriting recognition. K.S. and N.N. ensured the consistency of the work, supervised the publication process, and checked the text for content and language. All authors have read and agreed to the published version of the manuscript.

**Funding:** The work was supported by grant no. WZ/I-IIT/5/2023 from Bialystok University of Technology and funded with resources for research by the Ministry of Education and Science in Poland.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Publicly available datasets were analyzed in this study. This data can be found here: <http://etlcdb.db.aist.go.jp/>.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Velek, O.; Nakagawa, M. *Using Stroke-Number-Characteristics for Improving Efficiency of Combined Online and Offline Japanese Character Classifiers*; Document Analysis Systems V; Lopresti, D., Hu, J., Kashi, R., Eds.; Springer: Berlin/Heidelberg, Germany, 2002; pp. 115–118.
2. John, J.; Pramod, K.V.; Balakrishnan, K. Offline handwritten Malayalam Character Recognition based on chain code histogram. In Proceedings of the 2011 International Conference on Emerging Trends in Electrical and Computer Technology, Nagercoil, India, 23–24 March 2011; pp. 736–741. [\[CrossRef\]](#)
3. Zhu, B.; Nakagawa, M. A robust method for coarse classifier construction from a large number of basic recognizers for on-line handwritten Chinese/Japanese character recognition. *Pattern Recognit.* **2014**, *47*, 685–693. [\[CrossRef\]](#)
4. Zhou, X.D.; Wang, D.H.; Tian, F.; Liu, C.L.; Nakagawa, M. Handwritten Chinese/Japanese Text Recognition Using Semi-Markov Conditional Random Fields. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 2413–2426. [\[CrossRef\]](#) [\[PubMed\]](#)
5. Gayathri, P.; Ayyappan, S. Off-line handwritten character recognition using Hidden Markov Model. In Proceedings of the 2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Delhi, India, 24–27 September 2014; pp. 518–523. [\[CrossRef\]](#)
6. Plötz, T.; Fink, G.A. Markov models for offline handwriting recognition: A survey. *Int. J. Doc. Anal. Recognit. (IJDAR)* **2009**, *12*, 269. [\[CrossRef\]](#)
7. Zhou, X.D.; Zhang, Y.M.; Tian, F.; Wang, H.A.; Liu, C.L. Minimum-risk training for semi-Markov conditional random fields with application to handwritten Chinese/Japanese text recognition. *Pattern Recognit.* **2014**, *47*, 1904–1916. [\[CrossRef\]](#)
8. Liu, C.L.; Koga, M.; Fujisawa, H. Lexicon-driven segmentation and recognition of handwritten character strings for Japanese address reading. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 1425–1437. [\[CrossRef\]](#)
9. Zhang, X.Y.; Bengio, Y.; Liu, C.L. Online and Offline Handwritten Chinese Character Recognition: A Comprehensive Study and New Benchmark. *arXiv* **2016**, arXiv:1606.05763.
10. Solomon, C.; Breckon, T. *Fundamentals of Digital Image Processing: A Practical Approach with Examples in Matlab*; John Wiley & Sons: Chichester, UK, 2011.
11. Nixon, M.; Aguado, A.S. *Feature Extraction and Image Processing for Computer Vision*, 3rd ed.; Academic Press: Cambridge, MA, USA, 2012.
12. Jing, X.Y.; Chang, H.; Li, S.; Yao, Y.F.; Liu, Q.; Bian, L.S.; Man, J.Y.; Wang, C. Face Recognition Based on a Gabor-2DFisherface Approach with Selecting 2D Gabor Principal Components and Discriminant Vectors. In Proceedings of the 2009 Third International Conference on Genetic and Evolutionary Computing, Guilin, China, 14–17 October 2009; pp. 565–568. [\[CrossRef\]](#)
13. Dongcheng, S.; Fang, C.; Guangyi, D. Facial Expression Recognition Based on Gabor Wavelet Phase Features. In Proceedings of the 2013 Seventh International Conference on Image and Graphics, Qingdao, China, 26–28 July 2013; pp. 520–523. [\[CrossRef\]](#)
14. Zhang, Y.; Li, W.; Zhang, L.; Lu, Y. Adaptive Gabor Convolutional Neural Networks for Finger-Vein Recognition. In Proceedings of the 2019 International Conference on High Performance Big Data and Intelligent Systems (HPBD IS), Shenzhen, China, 9–11 May 2019; pp. 219–222. [\[CrossRef\]](#)
15. Buczkowski, M.; Szymkowski, P.; Saeed, K. Segmentation of Microscope Erythrocyte Images by CNN-Enhanced Algorithms. *Sensors* **2021**, *21*, 1720. [\[CrossRef\]](#) [\[PubMed\]](#)

16. Electrotechnical Laboratory, Japanese Technical Committee for Optical Character Recognition. ETL Character Database. 1973–1984. Available online: <http://etlcvdb.db.aist.go.jp> (accessed on 5 June 2023).
17. Jaeger, S.; Liu, C.L.; Nakagawa, M. The state of the art in Japanese online handwriting recognition compared to techniques in western handwriting recognition. *Int. J. Doc. Anal. Recognit.* **2003**, *6*, 75–88. [[CrossRef](#)]
18. Tabedzki, M.; Saeed, K.; Szczepański, A. A modified K3M thinning algorithm. *Int. J. Appl. Math. Comput. Sci.* **2016**, *26*, 439–450. [[CrossRef](#)]
19. Kato, N.; Suzuki, M.; Omachi, S.; Aso, H.; Nemoto, Y. A handwritten character recognition system using directional element feature and asymmetric Mahalanobis distance. *IEEE Trans. Pattern Anal. Mach. Intell.* **1999**, *21*, 258–262. [[CrossRef](#)]
20. Wakahara, T.; Kimura, Y.; Sano, M. Handwritten Japanese character recognition using adaptive normalization by global affine transformation. In Proceedings of the Sixth International Conference on Document Analysis and Recognition, Seattle, WA, USA, 13 September 2001; pp. 424–428. [[CrossRef](#)]
21. Tsuruoka, S.; Hattori, M.; Kadir, M.F.b.A.; Takano, T.; Kawanaka, H.; Takase, H.; Miyake, Y. Personal Dictionaries for Handwritten Character Recognition Using Characters Written by a Similar Writer. In Proceedings of the 2010 12th International Conference on Frontiers in Handwriting Recognition, Kolkata, India, 16–18 November 2010; pp. 599–604. . [[CrossRef](#)]
22. Gao, T.F.; Liu, C.L. LDA-Based Compound Distance for Handwritten Chinese Character Recognition. In Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), Curitiba, Brazil, 23–26 September 2007; Volume 2, pp. 904–908. [[CrossRef](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.