Journal of Soft Computing Exploration

Homepage: https://shmpublisher.com/index.php/joscex

p-ISSN: 2746-7686
e-ISSN: 2746-0991

.

Gabor wavelet and multiclass support vector machine for braille image classification

Feri Agustina[1], Eko Hari Rachmawanto[2*], Ni Kadek Devi Adnyaswari Putri[3], Fakhri Rasyid Saputro[4], Heru Lestiawan[5], Suprayogi[6], Solichul Huda[7]

[1,2,3,4,5,6]Study Program in Informatics Engineering, Universitas Dian Nuswantoro, Semarang, Indonesia
[7]Study Program in Broadcasting, Universitas Dian Nuswantoro, Semarang, Indonesia

Article Info

Article history:

Received September 5, 2024
Revised September 16, 2024
Accepted September19, 2024

Keywords:

Braille letter
Support vector machine
Gabor wavelet
Confusion matrix

ABSTRACT

Braille is a letter designed for the visually impaired. As a family with normal vision who have a visually impaired child find it difficult to Teach their child how to learn and understand the process of learning from home. Learning braille requires good finger sensitivity and memory to memorize each letterform, making it difficult to learn. With this study, braille letters can be detected from the image using the Gabor Wavelet method to extract braille images and combined with the Multiclass Support Vector Machine (Multiclass SVM) algorithm as a classification method for extracted braille images. Data testing was performed using a confusion matrix to determine the level of precision, accuracy, and recall. According to the results of tests performed on 910 braille data using confusion matrix, the highest recognition accuracy was 98,02%. The accuracy of these results is impacted by the parameters of the training process, the training data, and the test data used. This research has the opportunity to be developed in voice-based card recognition to help the visually impaired in the future research.

Corresponding Author:

Eko Hari Rachmawanto,
Faculty of Computer Science,
Universitas Dian Nuswantoro,
Jl. Imam Bonjol No.207, Pendrikan Kidul, Kota Semarang, Indonesia.
Email: eko.hari@dsn.dinus.ac.id
https://doi.org/10.52465/joscex.v5i3.474

1. INTRODUCTION

Blindness is a condition of a person who has impaired sense of sight. A visually impaired person is called visually impaired [1], visually impaired has two types of diseases, namely total blindness or total blindness and still has residual or low vision [2]. People with normal vision can exchange information through print and visual media. As for people with visual vision deficiencies or blindness, they have problems in how to exchange information in printed or written form. The visually impaired can receive information, one of which is by reading, but can only read with writing using braille [3], [4],. Braille letters have a pattern in the form of embossed dots totaling 6 points divided into three rows and two columns that can be read through the sense of touch, namely by using the tip of the finger [5].

Learning can be done at school and can also be done at home, while at home parents play an important role in the initial process of the child's education. In this case, parents play a role in helping children to be able to understand and understand their learning. However, for a person with a visual disability in reading and writing using braille, of course, most people who have normal vision do not understand braille [6]. This is what makes it difficult for parents to help visually impaired children in learning.

Up until now, braille letters have been translated by matching each braille letter to the corresponding alphabetical letter until the translation is accurate and legible. The process is not effective for everyone who is just learning braille. Thus, it is necessary to develop an intelligent system that is useful to help make it easier for ordinary people to recognize braille easily and effectively [7]. Braille research has been widely studied in various countries. The purpose of various braille studies is to facilitate braille learning for the visually impaired and people with normal vision. Therefore, it can help make it easier for everyone who does not know braille but deals with visually impaired people, especially parents who have visually impaired children. Thanks to the development of technology today, learning braille is easy, namely with the help of computer vision. Machine learning is currently the most widely used way as a tool to make it easier to develop algorithms that can learn and process model features from large amounts of data [8].

There are various approaches to developing a braille recognition system, including using different algorithms and extracting features. The extraction feature is used to be a parameter or value entered at the classification stage [9], [10]. Gabor Wavelet is one of the good extraction features in performing data extraction. Furthermore, the data that has been extracted will be classified. Support Vector Machine is one of the algorithms of various other classification algorithms that are often used in data processing when classifying data, this is because SVM is able to show good performance. In this study, in creating a braille recognition system, the Multiclass Support Vector Machine (Multiclass SVM) was used to classify each class category in its braille drawings. The two classes of data that are training data and test data are created separately aimed at maximizing distance.

According to research by Ibrahim et al. in 2018 [11] was conducted to develop a system for recognizing and translating braille images into letters that can be read by non-braille readers. The system used the bag of features (BOF) technique, which included the SURF feature as a descriptor, and the k-means clustering algorithm to determine the nearest clusters. The support vector machine was used as the classifier for braille images. The test results showed that the system had an accuracy of 97.44%. The goal of this study was to create a tool that could help beginners learn to read braille by providing a way to translate braille images into letters.

Sari et. al. in their research in 2017 used template matching to detect braille letters in real time [12]. The experiment was conducted with a detection time of 3 seconds, 5 seconds, 6 seconds and 10 seconds. The lowest accuracy obtained was at a detection time of 3 seconds of 92.3% and a 10-second waiting time with an accuracy of 100%. Unfortunately, a waiting time of 10 seconds is too long for the detection process per letter.

Research conducted by Ramadana [13] shows that android-based braille translator applications can help users learn and understand braille. In translating braille images into alphabet letters using the Convolutional Neural Network classification method using the MobileNetV2 architecture model consisting of 27 classes with a total of 8664 images, resulting in an accuracy level of 0.9183. Therefore, the performance of the application detection is not optimal or not accurate.

Herlambang et al, [14] have conducted Braille character recognition using the Convolutional Neural Network method with an accuracy rate of 81.54%, a precision value of 81.54%, a recall value of 100%, and an F1 score of 89.83% for a case study of Braille characters with image acquisition using a smartphone. and the slope level between 0 to 4 degrees with a distance of 30cm using a training model with a learning rate of 0.0001 and an adam optimizer.

Based on Shokat et al in 2020, only level 1 Braille data for English alphabets were used, and English characters corre-sponding to Braille input were recognized using DL techniques. Performance evaluation indicates that GoogLeNet Inception Model has the highest accuracy of 95.8% with95.89% specificity (TNR) and 99.83% sensitivity (TPR). It also has a reduced error rate of 3.39% [2].

Another research by Jha and Pavarthi in 2019, SVM for Character Recognition has been carried out for the process of recognizing handwritten Hindi braille. The proposed method first converts the scanned Hindi handwritten text into printable text using HOG features with SVM classifier and then maps the printable text to Braille using its UTF-8 code. he HOG features were also calculated assuming 4x4 cell size and overall accuracy rose to 95.56% as compared to 94.65% in the earlier case [9].

AlSalman et al in 2021 conducted on two datasets of Braille images to evaluate the performance of the DCNN model [5]. The first dataset contains 1,404 labeled images of 27 Braille symbols representing the alphabet characters. The second dataset consists of 5,420 labeled images of 37 Braille symbols that represent alphabet characters, numbers, and punctuation. The proposed model achieved a classification accuracy of 99.28% on the test set of the first dataset and 98.99% on the test set of the second dataset.

Here, we applied the extraction method from Gabor Wavelet and combined with the Multiclass Support Vector Machine (Multiclass SVM) method as a method to classify braille image objects. This study applies the Gabor Wavelet method as an extraction feature to enhance braille images to make it easier for braille characters to be identified with the Multiclass Support Vector Machine (Multiclass SVM) method. Next, braille characters that have been translated into letters of the alphabet.

2. METHOD

In this study, it requires material in the form of braille letters and requires a method to carry out the data extraction process and data classification to identify braille letters, namely by using Gabor Wavelet as an extraction feature and Multiclass Support Vector Machine (Multiclass SVM) as a data classification method.

2.1. Barille Letter

Braille letters are arranged consisting of 6 embossed points with a position of 3 vertical points and two horizontal points arranged in a matrix. The embossed point has a sequence number of 1-2-3, 4-5-6, as shown in Figure 1.

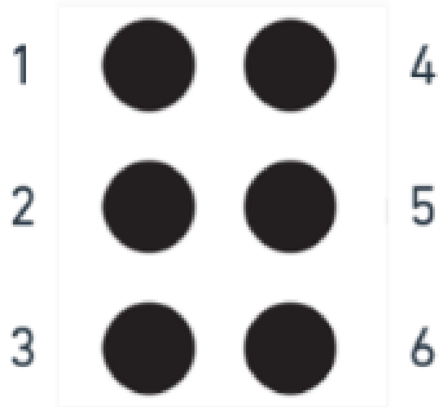

Figure 1. Braille embossed points

Each point can be formed and deleted to be able to create 64 braille character combinations. Braille combinations are of two types of combinations, namely one-sided and double-sided, as shown in Figure 2. Braille letters are read from left to right and can represent alphabetic letters, punctuation marks, numbers, musical notes, mathematical symbols, and more [15]. Braille letters have physical characteristics or cell specifications designed to identify braille characters. The commonly used Braille letters measure about 6 mm high and 4 mm wide and about 0.4 mm thick as shown in Figure 3. In mastering braille, there are several things that need to be considered including: understanding the location of points, touching sensitivity, understanding the position of lines. This ability is certainly difficult, so this research can help facilitate braille learning. Here is the form of braille characters to be recognized as in Figure 4.

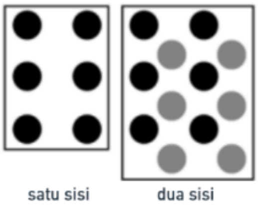

Figure 2. Braille cells

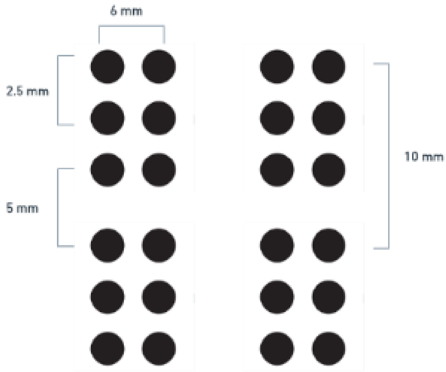

Figure 3. Braille specifications

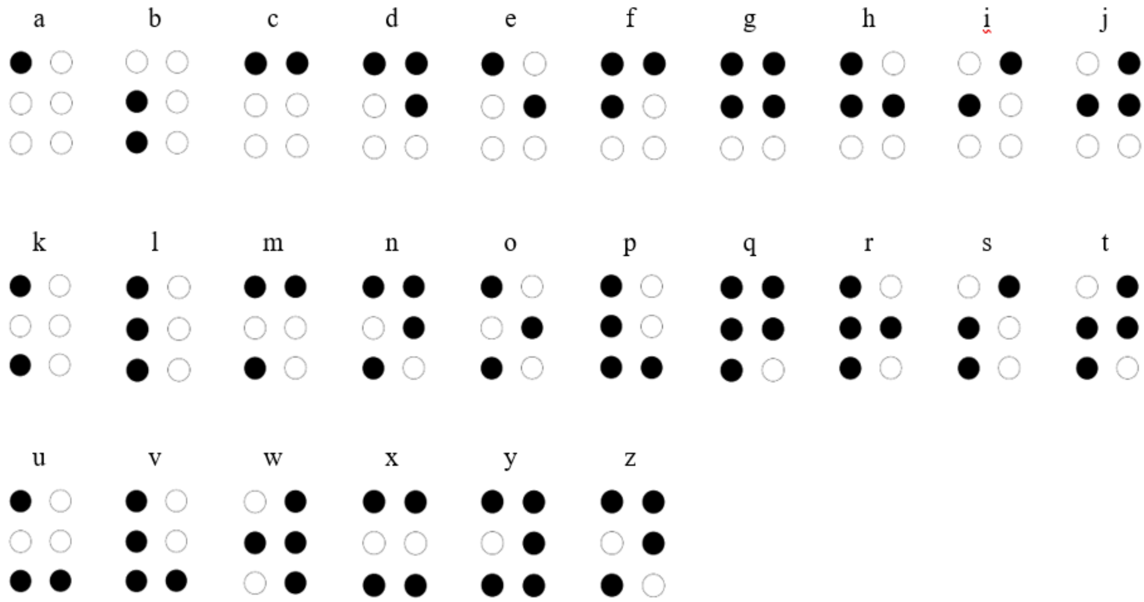

Figure 4. Braille letter

2.2. Research Stages

This study uses the stages of research as shown in the flowchart image in Figure 5. Here, we used dataset based on https://www.kaggle.com/datasets/shanks0465/braille-character-dataset. The first process is the process of inputting an image of braille letters in the program. Next, the image is saved in jpg format. Then there is a process that functions to improve the braille image which is carried out in the image processing process and prepares the image for the next process [16], namely the image segmentation process. In the segmentation process, the images on each word will be separated into images with each letter. After the images have been segmented into individual letters, the next step is to extract features from each image using the Gabor wavelet filter. The results of this feature extraction process are then stored in the training and test data sets. Once the data has been stored, the system is able to classify the images based on the extracted features using the multiclass support vector machine method. Finally, the system combines the identified letters into words based on the braille letters in the original image. This process is used to recognize and transcribe braille text from images.

.

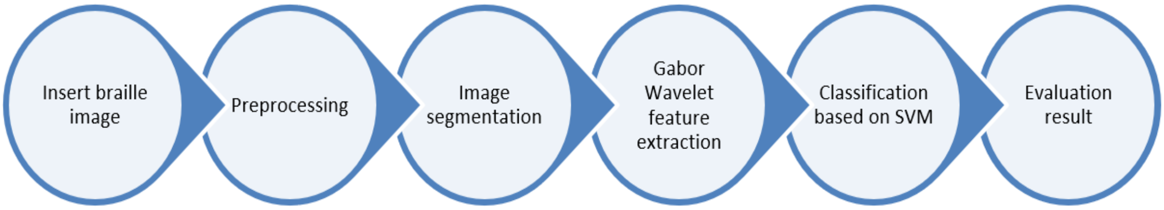

Figure 5. Stages of research

2.3. Image Processing

During the image processing stage, the image will be processed using image processing principles to extract the unique characteristics of each braille letter. These are the steps involved in this process:

1. Grayscale : Grayscale is a process that aims to convert image input into an image with a gray level color. Each image pixel that originally had a color with an RGB value, then the value is calculated to find the average value so that the color pixel value changes to a grayish value.
2. Median filter : The median filter is a technique used to remove noise from grayscale images. It works by calculating the median or middle value of the pixels surrounding a given pixel and replacing the value of the pixel with the calculated median value. This helps to smooth out any variations or noise in the image and can improve the overall quality of the image. The median filter is often used in image processing applications to pre-process images before they are further analyzed or used for other purposes.
3. Binary image : Binary Image is used to identify objects in the image, which are represented as regions in the image. To create a binary image, the pixel values of the original grayscale image are compared to the average gray value of the image. If a pixel value is greater than the average gray value, it is converted to white, or a binary value of 255. If it is less than the average gray value, it is converted to black, or a binary value of 0. This process helps to simplify the image and make it easier to analyze and recognize objects in the image.
4. Dilatation : Dilatation is the process used to add pixels to an image's border. This dilatation process aims to remove some small dots so that it will produce an image that only contains black dots. This process combines the dots of the image intersecting with the element structure and finds the maximum pixel value in the elemental structure environment.
5. Erosion : Erosion is the process used to make the black pixels more prominent or to remove pixels within the borders of a braille imageThis process will give the braille point size to its original size.
6. Canny Edge Detection : Canny Edge Detection is a process to extract the edges of objects in an image. This process is used to detect discontinuous brightness, extracting massive gradient brightness.
7. Draw Circle: Draw Circle is the process of drawing a circle on each edge detected in the previous process.
8. Inverse : Inverse or negation of an image is a process aimed at creating a negative image. In grayscale imagery the inverse process will change the pixels of white color to black, and vice versa the color black will become white. When viewed from a histogram, then a negative image is an image from mirroring the original image.

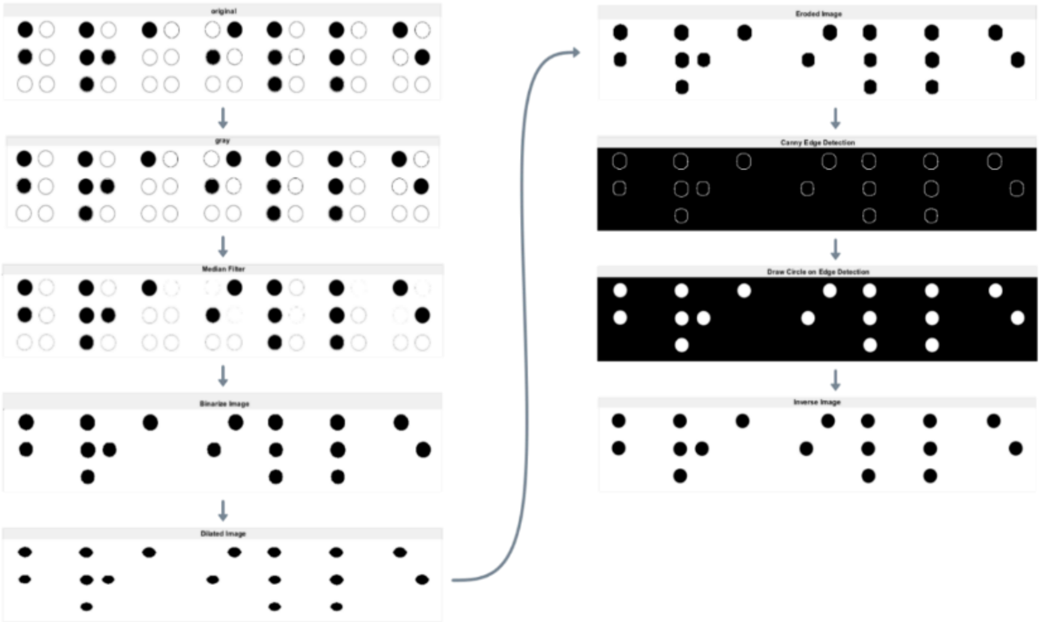

Figure 6. Results of image processing stages (a) Original image, gray, median filter, binary image, dilatation, (b) Erosion, canny edge detection, draw, inverse

2.4. Image Segmentation

In this process, images of braille text are first divided into individual letters and segmented. To do this, the dots of each braille letter are merged together using an erosion process with a kernel size of 100x7 pixels. The resulting image consists of black segments that represent letters and white space between letters. The next step is to detect the edges of the lines within the black segments and identify the coordinates of these lines [15], [17], [18]. Finally, the image is cropped based on the lines that have been detected. This results in an individual image for each letter that can be used for feature extraction.

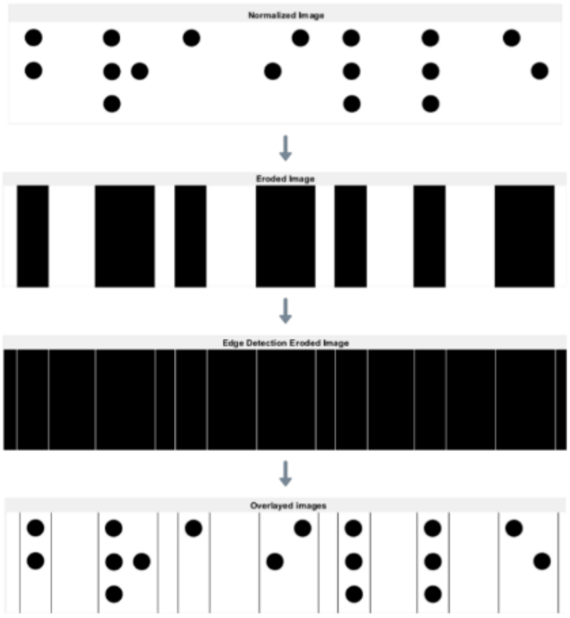

Figure 7. Results of the image segmentation stages

.

2.5. Gabor Wavelet Feature Extraction

Feature extraction is the process of extracting important information, such as certain characteristics of an object, from an image for storage and comparison. The Gabor wavelet is a type of filter that is often used for this purpose, as it is able to mimic the way the human visual system processes information by isolating specific frequencies and patterns from an image. This makes the Gabor wavelet particularly useful for facial image recognition applications that rely on identifying textures and patterns in images. To extract features from an image using the Gabor wavelet, the filter is applied to the image in order to highlight specific sizes, orientations, and shapes within the image. The resulting feature values can then be used as input for classification processes.

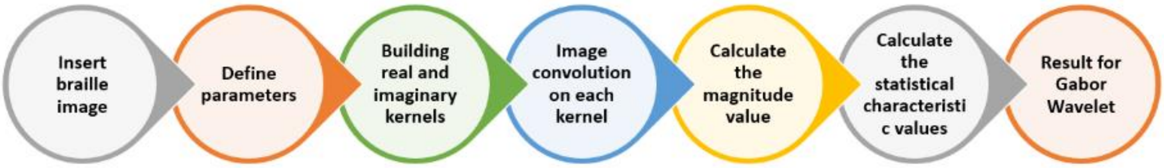

Figure 8. Gabor wavelet process

At the stage of the Gabor Wavelet extraction feature as in Figure 8, there are several processes carried out in extracting braille images, here are the steps:

- Image input : the process of inserting a pre-segmented image of braille. Furthermore, the image will be converted into a size that matches the size of all images that will be extracted.
- Determining parameters : In order to develop a Gabor wavelet filter, several parameters must be considered. These parameters include the wavelength of the filter, the angle of orientation of the filter relative to the image, the phase offset of the filter, the standard deviation of the Gaussian function used in the filter, the aspect ratio of the filter, and the ellipticity of the filter. All of these parameters play a role in determining the characteristics and functionality of the Gabor wavelet filter.
- Building real and imaginary kernels : the Gabor Wavelet process of constructing real and imaginary kernels requires the following formula.

$$g(x,y) = \frac{1}{2\pi\sigma_x\sigma_y}\exp\left(-\frac{1}{2}\left[\frac{x'^2}{\sigma_x^2}+\frac{y'^2}{\sigma_y^2}\right]\right)\cos\left(2\pi F x' + \psi\right) \quad (1)$$

$$g(x,y) = \frac{1}{2\pi\sigma_x\sigma_y}\exp\left(-\frac{1}{2}\left[\frac{x'^2}{\sigma_x^2}+\frac{y'^2}{\sigma_y^2}\right]\right) sin\left(2\pi F y' + \psi\right) \quad (2)$$

$$x' = x\cos\theta + y\sin\theta \quad (3)$$

$$y' = -x\sin\theta + y\cos\theta \quad (4)$$

$$mag = \sqrt{r^2 + im^2} \quad (5)$$

- Image convolution with each kernel : the process of generating real convolution images and imaginary convolution images of the multiplicity of images that each kernel has constructed.
- Calculating the magnitude value : this process of calculating is carried out using equation (5). The resulting value of magnitude consists of the square root of the sum of the squares of the real part and the square of the imaginary part.
- Calculating static characteristic values: this calculation process is used to perform the classification process. It involves calculating the mean, standard deviation, variance, and median statistics of values for the features to be extracted.

2.6. Support Vector Machine Classification

Support Vector Machine (SVM) is a *supervised learning* method used for classification [19](Support Vector Classification) and regression (Support Vector Regression). The basis of this method is to find the best hyperplane that acts as a separator of two classes on the input field. Hyperplane acts as the best separator for a set of positive (+) and negative (-) data classes by searching for hyperplane edges and maximum point values [9]. Margin definition is the summation of the distance between the hyperplane of each class with the nearest pattern or pattern. When the two classes can be separated linearly, SVM tries to find a separating hyperplane that gives the smallest generalization error among a number of different hyperplanes. In this case, the data closest to the hyperplane is called the support vector. Here's an example of defining the SVM hyperplane in Figure 9.

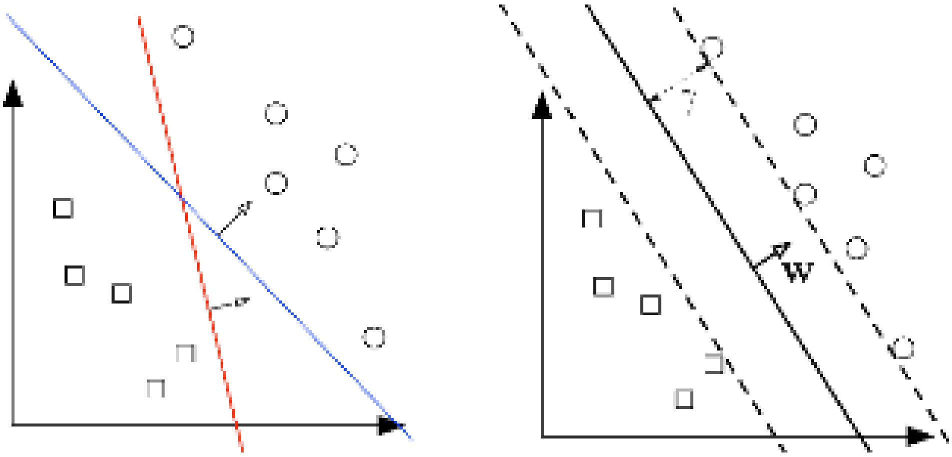

Figure 9. Hyperplane determination illustration

The Support Vector Machine was originally developed to classify only two classes. Then as it develops there are problems that cannot be solved by means of classification of two classes alone. Therefore, the researchers developed again to classify more than two classes or the so-called *multiclasses*. This *multiclass* classification is divided into two types, namely One Against All (OAA) and One Against One (OAO). The concept of the One Against All method is that one *classifier* is trained per class with one sample of that class classified as positive and all other samples as negative. At the same time, One Against One is a method that trains binary classification for K-way multi-class problems, each of which receives a sample of a pair of classes from the set at prediction, and must learn to distinguish between these two classes. Here is an illustrative image of the Multiclass SVM in Figure 10.

.

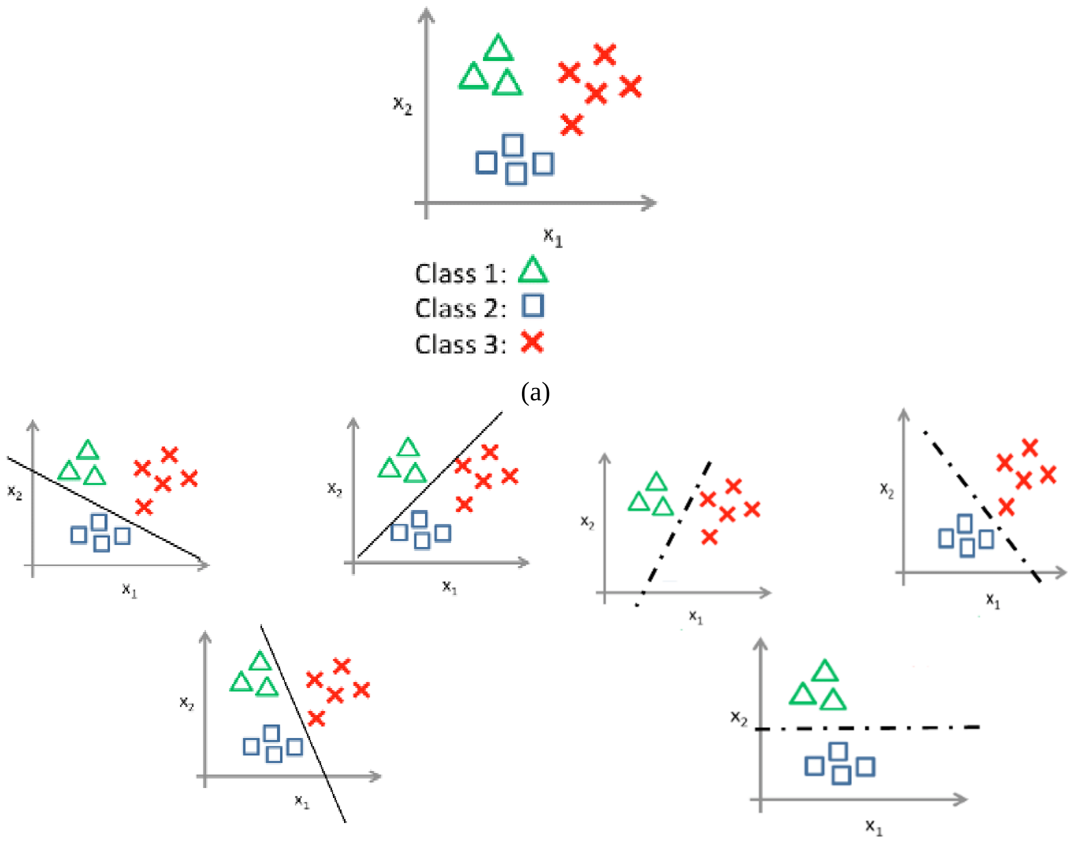

Figure 10. Multiclass SVM illustration : (a) Data multiclass SVM, (b) Multiclass SVM one against all (b) Multiclass SVM one against one

In this study, we used the One-vs-All method to determine the class of a new data point (x) based on the largest hyperplane value using the following equation (6).

$$class\ x = ar\ gmax\left(\sum_{i=1}^{m} a_i y_i\ K(x, x_i) + b\right) \tag{6}$$

The data extraction process that has been obtained will be used for the data classification process using the Multiclass Support Vector Machine method. This study conducted a classification on 26 data from the letters of the alphabet. In the use of Multiclass SVM will use the One Against All method based on Radial Basis Function (RBF) kernel. The first process is to read the feature extraction and select the optimal feature values to perform the classification. It then performs the formatting of the train data and the test data and class labels. In the classification process, it is necessary to determine the optimal parameters and then adjust the model to the training data. Additionally, the model can be used to predict test data based on selected features. The outcome of the classification process is a class label for each feature that was tested.

2.7. Testing method

The testing phase is used to determine the success rate in the data classification system that has been built. This study implements system testing by creating multiclass confluence matrices and validating the data that has been tested on the system. Accuracy is a measure of how well the system is able to classify data points correctly [20]. It is calculated by dividing the number of correct predictions made by the system by the total number of predictions as in (7).

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{7}$$

3. RESULTS AND DISCUSSIONS

Based on the results of the implementation and testing that has been carried out previously, it is proven that the Gabor Wavelet extraction feature can work optimally together with the Multiclass Support Vector Machine algorithm. The result of identifying braille letters can obtain a good accuracy value. This is supported by the results of the confusion matrix testing, which yielded an accuracy value of 99,12%. Table 2 shown the comparison of accuracy based previous research.

Table 2. The Comparison results

No	Research by	Year	Algorithm	Accuracy
1	Sari, et. al.	2017	Template matching	92,3%
2	Ibrahim, et. al.	2018	Bag of Features (BOF) + SVM	97,44%
3	Jha and Pavarthi	2019	SVM	94%
4	Shokat, et. al.4	2020	CNN GoogLeNet Inception	95,80%
5	AlSalman, et al.	2020	Deep convolutional neural network	98,99%
6	Herlambang, et al.	2020	CNN	81,545
7	Ramadana and Selao	2024	CNN MobilenetV2	91,83%
			MultiSVM only 882-910	96,92%
8	Our proposed	2024	Gabor wavelet + MiltiSVM (RBF kernel) - 902/910	99,12%

4. CONCLUSION

This research shows that the Gabor Wavelet extraction feature and the Multiclass Support Vector Machine algorithm are successfully implemented and can work well in identifying braille images into alphabetic letters in digital image processing. Gabor wavelet used as feature extraction to helps improved accuracy before apply the Multiclass Support Vector. In identifying using braille in uppercase and lowercase letters on tests from the accuracy values of 99,12%. This study used test data of 910 randomly distributed data. Inaccurate classification results can be caused by stains on the paper and the slope of braille writing on the image. There are several limitations to this study that could be addressed in future research. Some possible areas for improvement include the ability to recognize braille letters, numbers, and punctuation, the ability to identify sentences in braille, and the ability to differentiate between braille dots and other marks on paper. These are all areas that could be explored in order to further develop and improve the performance of the application.

REFERENCES

[1] M. R. Ningsih *et al.*, "Sign Language Detection System Using YOLOv5 Algorithm to Promote Communication Equality People with Disabilities," *Sci. J. Informatics*, vol. 11, no. 2, hal. 549–558, 2024, doi: 10.15294/sji.v11i2.6007.

[2] S. Shokat, R. Riaz, S. S. Rizvi, A. M. Abbasi, A. A. Abbasi, dan S. J. Kwon, "Deep learning scheme for character prediction with position-free touch screen-based Braille input method," *Human-centric Comput. Inf. Sci.*, vol. 10, no. 1, 2020, doi: 10.1186/s13673-020-00246-6.

[3] I. Sulviyani dan L. Novamizanti, "Perancangan Konversi Braille Ke Teks Berbasis Android," *ReTII*, 2017.

[4] R. Patmasari, S. Saidah, A. F. Akbar, dan R. Magdalena, "Design and Implementation of Learning Tools To Read the Braille Letters Based on Voice Processing and Arduino Using Mel Frequency Cepstral Coefficient and K-Nearest Neighbor Method," *J. Meas. Electron. Commun. Syst.*, vol. 6, no. 1, hal. 28, 2020, doi: 10.25124/jmecs.v6i1.2019.

[5] A. AlSalman, A. Gumaei, A. AlSalman, dan S. Al-Hadhrami, "A Deep Learning-Based Recognition Approach for the Conversion of Multilingual Braille Images," *Comput. Mater. Contin.*, vol. 67, no. 3, hal. 3847–3864, 2021, doi: 10.32604/cmc.2021.015614.

[6] I. G. Ovodov, "Optical Braille Recognition Using Object Detection Neural Network," *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2021-October, hal. 1741–1748, 2021, doi: 10.1109/ICCVW54120.2021.00200.

[7] R. Goebel, "PRICAI 2019 : Trends in Series Editors," *Pricai*, vol. 2, no. 61672281, hal. 431–444, 2019, doi: 10.1007/978-3-030-29894-4.

[8] A. Iswara, A. M. Iswara, M. H. Purnomo, dan S. Sumpeno, "Aplikasi Pengenalan Kata pada Huruf Braille dan Pelafalannya," *J. Tekn Ik Pomits*, vol. 2, no. 2, hal. A335–A340, 2013, [Daring]. Tersedia pada: http://ejurnal.its.ac.id/index.php/teknik/article/view/3408.

[9] V. Jha dan K. Parvathi, "Braille transliteration of hindi handwritten texts using machine learning for character recognition," *Int. J. Sci. Technol. Res.*, vol. 8, no. 10, hal. 1188–1193, 2019.

[10] R. F. Rahmat *et al.*, "Braille letter recognition in deep convolutional neural network with horizontal and vertical projection," *Bull. Electr. Eng. Informatics*, vol. 13, no. 5, hal. 3380–3391, 2024, doi: 10.11591/eei.v13i5.7222.

[11] S. Ibrahim, N. A. Tarmizi, N. Sabri, N. F. Mohd Johari, dan A. F. Ahmad Fadzil, "Braille image recognition for beginners," *2018 9th IEEE Control Syst. Grad. Res. Colloquium, ICSGRC 2018 - Proceeding*, no. August, hal. 81–84, 2018, doi: 10.1109/ICSGRC.2018.8657490.

[12] D. P. Sari, S. Rasyad, E. Evelina, dan A. Amperawan, "Identifikasi Huruf Braille Berbasis Image Processing Secara Real Time," *J. Ampere*, vol. 2, no. 2, hal. 68, 2017, doi: 10.31851/ampere.v2i2.1765.

[13] N. Azizah Ramadana dan A. Selao, "Aplikasi Penerjemah Huruf Braille Berbasis Android," *J. SINTAKS Log.*, vol. 4, no. 2, hal. 2775–412, 2024, [Daring]. Tersedia pada: https://jurnal.umpar.ac.id/index.php/sylog.

.

[14] M. F. Herlambang, A. N. Hermana, dan K. R. Putra, "Pengenalan Karakter Huruf Braille dengan Metode Convolutional Neural Network," *Syst. Inf. Syst. Informatics J.*, vol. 6, no. 2, hal. 20–26, 2021, doi: 10.29080/systemic.v6i2.969.

[15] R. Li, H. Liu, X. Wang, J. Xu, dan Y. Qian, "Optical braille recognition based on semantic segmentation network with auxiliary learning strategy," *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work.*, vol. 2020-June, hal. 2362–2368, 2020, doi: 10.1109/CVPRW50498.2020.00285.

[16] J. Unjung dan M. R. Ningsih, "Optimized Handwriting-based Parkinson ' s Disease Classification Using Ensemble Modeling and VGG19 Feature Extraction," *Sci. J. Informatics*, vol. 10, no. 4, hal. 489–498, 2023, doi: 10.15294/sji.v10i4.47108.

[17] H. Hadiq, S. Solehatin, D. Djuniharto, M. A. Muslim, dan S. N. Salahudin, "Comparison of the suitability of the otsu method thresholding and multilevel thresholding for flower image segmentation," *J. Soft Comput. Explor.*, vol. 4, no. 4, hal. 242–249, 2023, doi: 10.52465/joscex.v4i4.266.

[18] A. A. Hakim dan E. Juanara, "Mask Detection System with Computer Vision-Based on CNN and YOLO Method Using Nvidia Jetson Nano," *J. Inf. Syst. Explor. Res.*, vol. 1, no. 2, hal. 109–122, 2023.

[19] E. Pirdia Wanti, A. Pariyandani, S. Zulkarnain Syed Idrus, dan A. Hasudungan Lubis, "Utilization of SVM Method and GLCM Feature Extraction in Classifying Fish Images with Formalin," *Sci. J. Informatics*, vol. 8, no. 1, hal. 168–175, 2021, doi: 10.15294/sji.v8i1.26806.

[20] M. R. Ningsih, "Classification Email Spam using Naive Bayes Algorithm and Chi-Squared Feature Selection," vol. 9, no. 1, hal. 74–87, 2024.