JOURNAL OF DATA SCIENCE
eISSN:2805-5160
Vol.2022:14

Utilizing Support Vector Machines to Recognize Pattern Numbers on Sheet C1 Ballots

David Agustian[1], Tri Basuki Kurniawan[2*], Misinem[3], Edi Surya Negara[1], Yesi Novaria Kunang[2]

[1]Magister Teknik Informatika, Universitas Bina Darma, Palembang, Sumatera Selatan
[2] Sistem Informasi, Fakultas Ilmu Komputer,Universitas Bina Darma, Palembang, Sumatera Selatan
[3] Teknik Komputer, Fakultas Vokasi, Universitas Bina Darma, Palembang, Sumatera Selatan

[*]**Email**: tribasukikurniawan@binadarma.ac.id

Abstract

The election vote-counting process can be done automatically by applying machine learning technology, namely computer vision which is usually used for the pattern recognition process in images and videos. In this study, the object of pattern recognition is the C1 ballots for the 2019 simultaneous elections. The dataset used is the latest C1 format which was made by the author himself. To distinguish each number, C1 ballots will be colored to interpret a number. Then the numbers were captured using an android camera and saved in jpeg format. Furthermore, the author uses the python skimage module and GIMP for image processing and divides the dataset into 80% as training data and 20% as testing data. The model used in this study is the Support Vector Machine algorithm to classify images. The results that can be reached by this model were 100% accurate.

Keywords

Support Vector Machine, Machine Learning, Image Recognition, C1 Ballot

Introduction

Citing a news article by (Kumparan News, 2019) On April 17, 2019, Indonesian residents made political history as they held simultaneous elections for the first time. In this election, the President, DPR RI, DPD RI, Provincial DPRD, and Regency/City DPRD are all combined into one category. A case brought against Law (UU) No. 42/2008 governing the election of the President to the Constitutional Court served as the catalyst for this incident (MK). The lawsuit was then approved by the Constitutional Court, which also made a ruling on the law under review. The tragedy of the execution of this event, which resulted in the deaths of hundreds of officers, lies behind it. This tragedy is believed to be the result of officer tiredness brought on by an imbalanced workload. One of the reasons for this weariness is that, in accordance with the election rules, the process of recapitulation of c1 ballot papers still uses a manual and tiered approach starting from the TPS, sub-district, district/city, province, and national levels. This prolongs the process duration.

Submission: 16 November 2022
Acceptance: 28 November 2022

Based on the aforementioned occurrences, this research presents a strategy to use pattern recognition and machine learning techniques to automate the manual vote-counting procedure. The Support Vector Machine is one of the strategies employed (SVM). The optimum hyperplane that separates two data classes is what the SVM concept seeks. A hyperplane is a line that separates two data classes into two portions, with each data class present on both sides of the line. Additionally, the C1 sound sheet data will be reorganized for use in this investigation. Five columns and seven rows make up the new format, which will be colored to represent the digits 0 through 9. In its application, this new format will take the shape of visual data with dimensions of 15 by 21 pixels and colors of different hues.

Each number will have a different pixel value with these various color kinds. Additionally, it is what sets each number apart. The accuracy in detecting the pattern of each number should then be as expected in the goal of this study and be able to speed up the counting of ballots by utilizing a support vector machine model and a dataset in this new format. because, as shown in the research example below, the support vector machine model was able to provide accuracy above 90% in earlier investigations.

In a study by (Gorgevik & Cakmakov, 2001), a feature extraction technique that combined the SVM model and rule-based reasoning was able to extract features from a grayscale picture object with 16x16 pixel dimensions with an average accuracy of 97.65%. Then, in a different study (Tuba et al., 2016), handwritten number recognition was discussed. This was done using the support vector machine approach for object classification and was then optimized using the bat algorithm. A normal MNIST database with a size of 28x28 pixels and no image preprocessing was utilized as the dataset for testing this model. Applying this strategy yields an accuracy of 95.60%. In a study by (Burges & Schölkopf, 1997), the error rate in the pattern recognition process was reduced from 1.4% to 1.0% using 10,000 numerical data points obtained from NIST. The speed of the process was also improved, and in this study, it was noted that it was 22 times faster than the standard support vector machine method. The goal of this is to solve the method's invariance issues.

They investigated handwritten number recognition using a support vector machine for item categorization in their study (Tuba et al., 2016), and they optimized utilizing the bat algorithm. It is hoped that by employing the MNIST database as an object, the optimized support vector machine will produce results with a high degree of accuracy. All objects are converted to 28x28 pixels, with 60,000 image data used for training and 10,000 for testing. Applying this strategy yields an accuracy of 95.60%.

Pattern recognition

Patterns are closely related to things that have a variety of qualities. For instance, the ratio between lip height and lip height, the center distance between the left and right eyes, and the width of the lips are all characteristics of the face as an object. the ratio's value, which represents one of an object's defining parameters. While the concept of "recognition" refers to the classification of an object based on the traits it possesses. Thus, according to (Arif Muntasa, 2015), pattern recognition can be thought of as a method for automatically identifying objects based on traits learned from training data.

Submission: 16 November 2022
Acceptance: 28 November 2022

Starting with the detection/segmentation process, the extraction process, and the similarity measurement process or the recognition process, pattern recognition is a sequence of sequential procedures. (Yodha & Kurniawan, 2014) claimed in their journal article that pattern recognition is an artificial intelligence field that seeks to categorize objects based on the measurement of their properties. In another description, pattern recognition is described as a discipline whose goal is to categorize or classify objects (Kusmadewi, 2003).

Feature Extraction

Feature extraction is described as a procedure to highlight features and reduce the dimensions of objects—in this case, image objects—from high dimensions to lower dimensions (Mauridhi Hery & Arif Muntasa, 2010)). With this method, a label will be created using the most pertinent features.

Support Vector Machine

A support vector machine is a device that uses vectors as supports or markers to divide data into two groups, according to the article (MB Herlambang, 2019). To put it another way, SVM is a methodology or procedure that employs two points or two vectors, which together form a boundary line (or side limit for data that has 3 or more classes). A hyperplane is a boundary produced by these two points. This method seeks the optimal separation between the input data points in N-dimensional space (number of N-features). Support vector machines are one type of supervised machine learning method, (Kowalczyk, 2017). The goal of this supervision is to ensure that the data used to train and test the model already has a label indicating the desired outputs. similar to Figure 1 below.

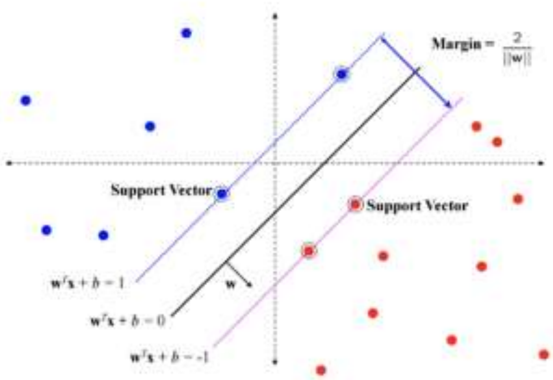

Margin = $\frac{2}{||\mathbf{w}||}$
Support Vector
Support Vector
$\mathbf{w}^T\mathbf{x} + b = 1$
$\mathbf{w}$
$\mathbf{w}^T\mathbf{x} + b = 0$
$\mathbf{w}^T\mathbf{x} + b = -1$

Figure 1. The Workings of SVM Classification

SVM is able to recognize data patterns that include multi classes by linking several binary SVM classifiers, despite the fact that SVM is developed for data classification that has two classes (binary mode). The SVM binary classifier's multi-class problem is frequently solved using the "One Versus One" and "One Versus All" techniques. The classifier set for each pair of classes in the One Versus One technique will be divided into two classes. On the other hand, classifiers are set for each type and various segmentation classes from other classes in the One Versus All system. Due to its simplicity, One Versus All is frequently used for identification. Kernel functions are used for data that has several classes. The linear kernel is one of the most basic kernel functions, while the radial basis (RBF) kernel function is the most popular, according to (Salcedo-Sanz et al., 2014).

Methodology

Data Gathering

The author of this study used the descriptive analysis method when looking at the KPU C1 ballot sheet. According to the book (Drs. Ulber Silalahi, 2014), the descriptive technique is a strategy for solving problems that entail defining the condition of the subject and object in the study, which can be people, institutions, or communities, based on what they are or how they are today.

Data Identification

The identification stage involves identifying the issues that the research topic has, which leads to the formulation of problems that are specific to the research to be done. The KPU C1 ballot sheet's official format is examined during the identification step, after which a new format for the C1 ballot sheet that will be used for this research is designed. The original C1 ballot form from the KPU is referenced in the new version of the C1 ballot sheet that will be used in this study. Figure 2 depicts the format of the original document.

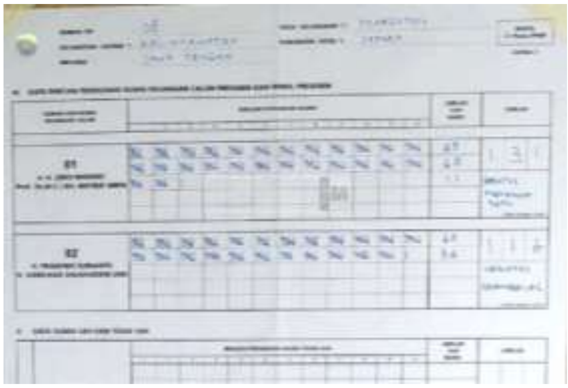


Figure 2. Original C1 Ballot Sheet

As seen in the amount column of Figure 2 above, the new format will be suggested for this column. As seen in figure 3, this new structure consists of three columns, each of which is composed of squares.

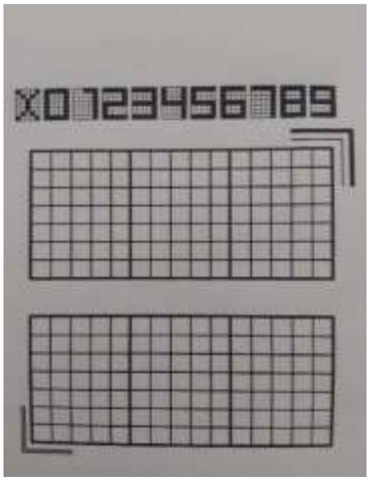


Figure 3. Proposed New Format of Sheet C1

Each column is 7x5 in size, and a thicker border separates each column from the others. The column is then colored to represent the digits 0 through 9, as well as the symbol "X," which will be used as a data indication that isn't a number.

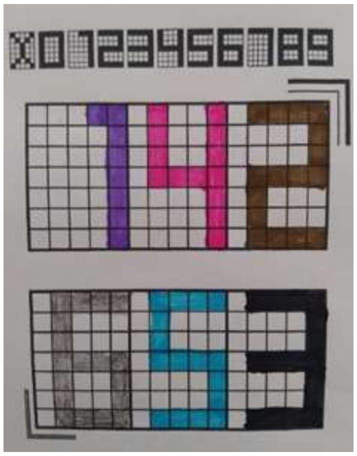


Figure 4. Coloring Guide

As seen in Figure 4 above, each column should be colored according to certain rules. Giving these varied color schemes seeks to produce diverse pixel values since these pixel values will develop into patterns in the computer's recognition of numbers. Using an image editor program, such as GIMP, the data will be cropped with a size of 15x21 pixels and turned into grayscale after coloring and preprocessing. Additionally, all of the cut numbers are saved as picture data in the.jpeg,.jpg, or.png formats. In contrast, this procedure is also done using scikit-image in Python programming in addition to the GIMP application for data pretreatment.

The author now gathers samples of the newly formatted C1 sheets that have been colored by photographing them with an Android camera in as many as 329 samples, then saving them in.jpg or.png form as a dataset. Figure 5 below shows the outcomes of this collection of numerical data.

Submission: 16 November 2022
Acceptance: 28 November 2022

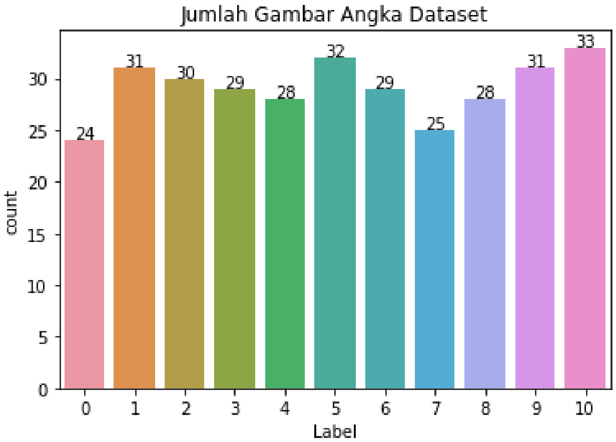


Figure 5. Coloring Guide

Following this, data processing using a ski Image takes place. The following are some of these procedures:

1. Scale the photo down to 15x21 pixels.
2. Change the picture's color to gray (grayscale). The varied quality and brightness of the acquired image will provide a bad image as a result.
3. Use Unsharp masking to sharpen the image.
4. Using the top-hat filter, remove any little items from the image.
5. Increase the picture data's threshold.

The principal component analysis (PCA) method, an eigenvector-based multivariate analysis that is frequently used to extract the best data variance, will be employed for feature extraction prior to the model identification phase. When a pattern in the data is found, the data can be reduced with minimal information loss. This is the main advantage of PCA.

Flowchart

According to Figure 6, this study will go through the following phases:

1. Resizing, grayscaling, unsharp masking, a white top-hat filter, and thresholding are all examples of pre-processing.
2. Feature extraction using PCA.
3. SVM-based classification.

Submission: 16 November 2022
Acceptance: 28 November 2022

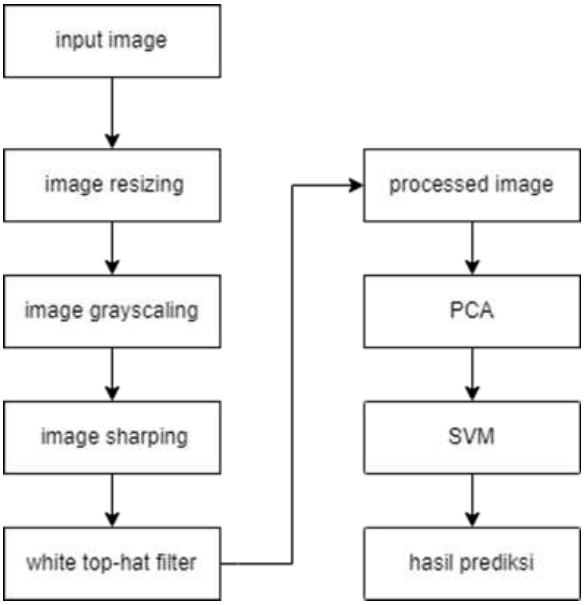


Figure 6. Flowchart

Results and Discussion

There are 2 image data processing processes, namely by using a skiimage and using the GIMP 2.10.30 application. Then the accuracy of model detection against the results of these two data will be compared as an evaluation of the model's performance. The following is the original image of the dataset that will be used in this study, as shown in Figure 7.

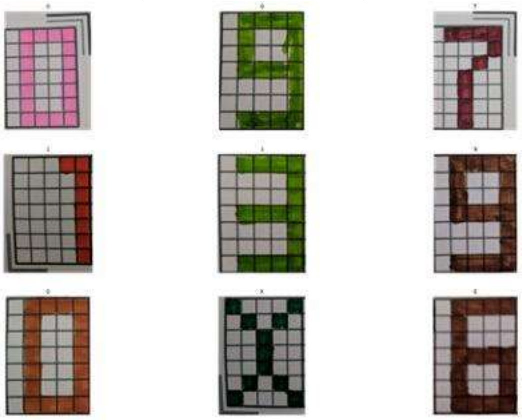


Figure 7. Research Dataset

Resize

Using the transform module resize() method of scikit-image, resize an existing image. (Drs. Ulber Silalahi, 2014) states that this method will process the image into a numpy array, convert it to the desired output form, and scale it to a specific size. The image data for this investigation will be scaled down to 15 × 21 pixels. As seen in Figure 8, below are several instances of altered photographs.

Submission: 16 November 2022
Acceptance: 28 November 2022

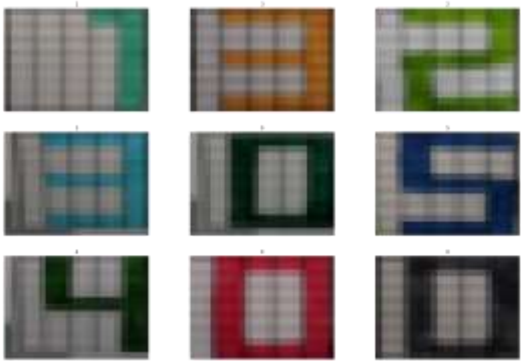


Figure 8. Image Size 15 x 21 Pixels

Grayscaled

The anticipated input image is a two-dimensional vector, or a grayscale image, before filters and other processing methods are used. Use the rgb2gray module from the skimage package to convert color image data (RGB), which has three channels, to gray image data, which has one channel. Skimage.color.rgb2gray() is the function used to convert RGB images to grayscale format (Van Der Walt et al., 2014).

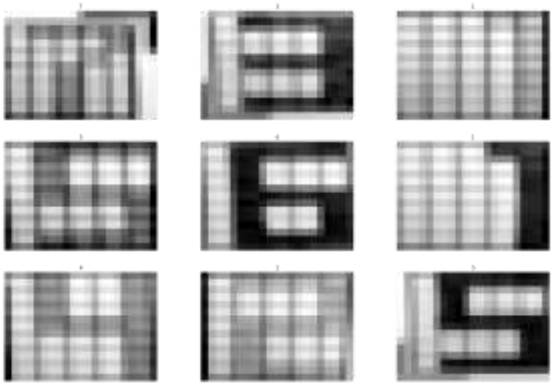


Figure 9. Convert Image to Grayscale

Unsharp Masking

This classification procedure heavily relies on the white top-hat filter and unsharp masking. The properties of each image with its many colors have a significant impact on the model's accuracy. Utilizing an image that has been slightly blurred, the sharpening procedure operates. An unsharp mask is then created by subtracting this from the original to identify any edges. Using this mask, contrast is then carefully enhanced along these edges, sharpening the final image (Polesel et al., 2000). The amount, radius, and threshold are used to govern unsharp masking.

1. The quantity governs the size of each overshoot and is expressed as a percentage (how darker and lighter the outline will be). The amount of contrast provided to the edges is also taken into account.
2. A lesser radius value will add detail with a smaller scale, however, a large radius value will produce halos around the edges that can be detected as objects. Radius controls the size of the edge to be increased or how broad the edge of the edge will be formed.

Submission: 16 November 2022
Acceptance: 28 November 2022

3. Threshold regulates minute adjustments in the brightness level that need to be sharpened. Threshold settings can be used to smooth off rough edges while sharpening more pronounced edges.

The skiimage.filters.unsharp mask() module is used in this study's use of unsharp masking, with radius = 10 and amount = 2. Here are some illustrations of photos with unsharp masking applied.

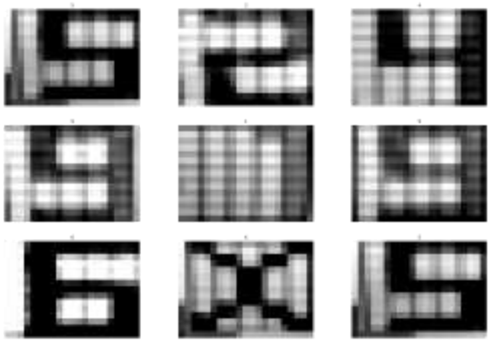


Figure 10. Image After Sharpen

White Top-hat Filter

The white top-hat of an image can be interpreted as the original\morphology\opening\ image of the image. The morphology of the opening of an image is the operation of erosion followed by dilation. The erosion operation is to reduce the number of pixels at the boundary between objects in one image which has the following rules for grayscale images, the value of the operation result (output pixels) is the minimum value obtained from the set of neighboring pixels (Guerroudji & Ameur, 2016). In a binary image, if there is a neighboring pixel that has a value of 0 then the output pixel will be changed to 0. The dilatation operation will add pixels at the boundary between objects in one image which has the following rules for grayscale images, the value of the operation result (output pixels) is the maximum value. obtained from the set of neighboring pixels. In a binary image, if there is a neighboring pixel with a value of 1, the output pixel will be changed to 1 (Bai et al., 2012).

This operation results in bright pixel dots that are smaller than the footprint. The footprint value is obtained from the function skiimage.morphology.disk() with a value of radius = 1.5. To apply the white top-hat filter use the function skiimage.morphology.white_tophat() which will return the value of image\ original-image\ white\ tophat. Here are some sample images after applying the white top-hat filter, as shown in Figure 11 below:

Submission: 16 November 2022
Acceptance: 28 November 2022

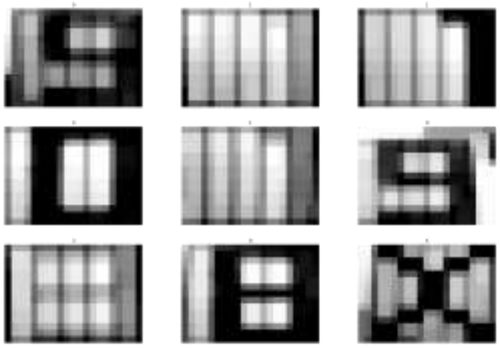


Figure 11. Image After Application of White Top-hat Filter

Threshold Otsu

Pixels are divided into two classes, foreground, and background, using the threshold otsu technique. By decreasing or enhancing the intensity between classes, the threshold value is established. The threshold algorithm operates as follows:

1. Input image
2. Get the histogram of the image
3. Calculate the threshold value T
4. Change the image pixels to white in certain areas, where the saturation is greater than T and vice versa change the image pixels to black, where the saturation is greater than T.

By minimizing the variance of each class, the object is segmented at the Otsu threshold in the image histogram procedure. Bimodal images respond nicely to this technique. Two peaks with a wide range of intensity levels may be seen in the image's histogram. The basic goal of this algorithm, as illustrated in Figure 12 below, is to divide the image histogram into two groups with a predetermined threshold by minimizing the weight variance of each class (Otsu et al., 1979).

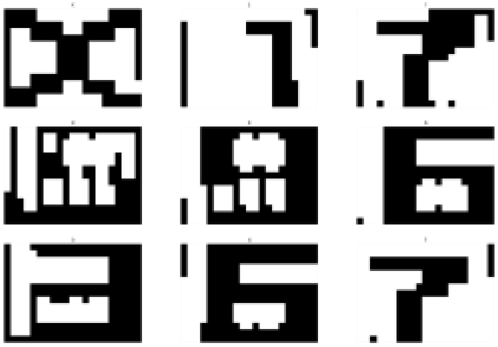


Figure 12. Image After Application of Threshold Otsu

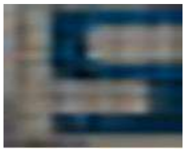
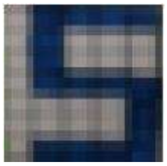
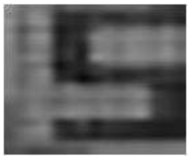
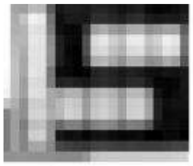
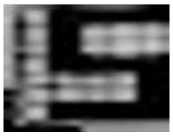
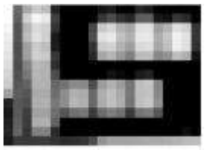
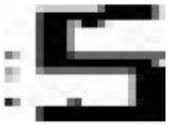
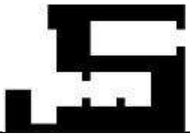
Image Processing using GIMP 2.10.30

The processes that are completed in manual image processing using GIMP 2.10.30 are nearly identical to those that are completed in image processing using the prior schema. Each piece of data will undergo resizing, grayscale conversion, Unsharp masking, and thresholding. It's only that

Submission: 16 November 2022
Acceptance: 28 November 2022

this technique does not employ a white top-hat filter and instead determines the threshold automatically. Table 1 below shows a comparison of the outcomes of these two processes:

Table 1. Comparison of GIMP and Skimage Image Processing Results

Image Processing	GIMP	Skimage
Image Size 15 x 21 Pixels		
Grayscaling		
Unsharp masking		
Thresholding		

Feature Extraction / PCA

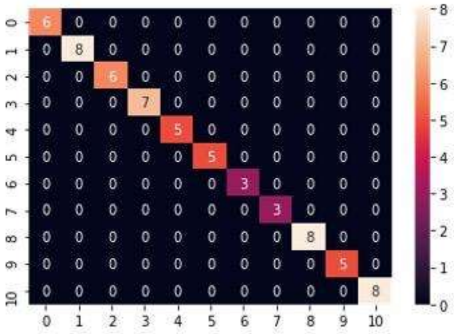
PCA is a multivariate analysis method that uses eigenvectors to extract the optimal data variance. Utilizing correlations between particular dimensions, this technique seeks to give the fewest possible variables while preserving the distribution of the original data (Kunaver & Tasič, 2005). PCA was employed in this study to reduce the size of the images. Numerous prior studies, including those conducted by (Lu, n.d.), (Rizanti et al., 2016), and (Lu, n.d.) have combined the PCA feature extraction approach and the support vector machine method (Zhao et al., 2017).

Image Recognition using SVM Model

The model's accuracy on testing data obtains 100% accuracy from the training process of picture data that is processed with a schematic and model parameters using kernel = RBF, C = 100, and gamma = 0.001. The accuracy is only 89% correct when using the GIMP 2.10.30 application and the same model settings on image data.

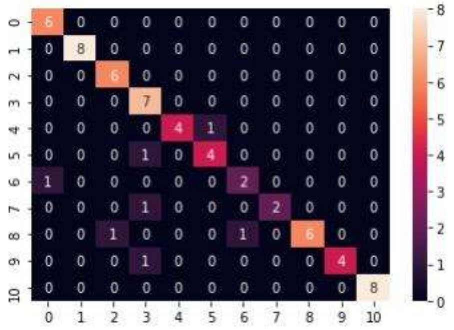
Model Evaluation

Figures 13, 14, 15 and 16 illustrate the categorization report and confusion matrix results, respectively.



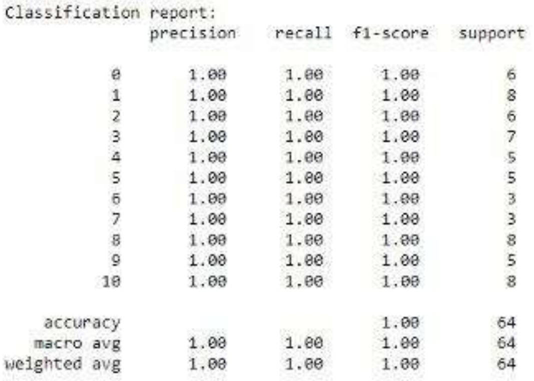
	0	1	2	3	4	5	6	7	8	9	10
0	6	0	0	0	0	0	0	0	0	0	0
1	0	8	0	0	0	0	0	0	0	0	0
2	0	0	6	0	0	0	0	0	0	0	0
3	0	0	0	7	0	0	0	0	0	0	0
4	0	0	0	0	5	0	0	0	0	0	0
5	0	0	0	0	0	5	0	0	0	0	0
6	0	0	0	0	0	0	3	0	0	0	0
7	0	0	0	0	0	0	0	3	0	0	0
8	0	0	0	0	0	0	0	0	8	0	0
9	0	0	0	0	0	0	0	0	0	5	0
10	0	0	0	0	0	0	0	0	0	0	8

Figure 13. Confusion Matriks Skimage



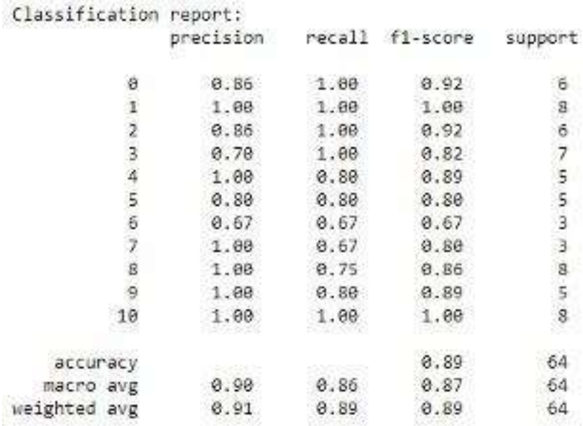
	0	1	2	3	4	5	6	7	8	9	10
0	6	0	0	0	0	0	0	0	0	0	0
1	0	8	0	0	0	0	0	0	0	0	0
2	0	0	6	0	0	0	0	0	0	0	0
3	0	0	0	7	0	0	0	0	0	0	0
4	0	0	0	0	4	1	0	0	0	0	0
5	0	0	0	1	0	4	0	0	0	0	0
6	1	0	0	0	0	0	2	0	0	0	0
7	0	0	0	1	0	0	0	2	0	0	0
8	0	0	1	0	0	0	1	0	6	0	0
9	0	0	0	1	0	0	0	0	0	4	0
10	0	0	0	0	0	0	0	0	0	0	8

Figure 14. Confussion Matriks GIMP



```
Classification report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00         6
           1       1.00      1.00      1.00         8
           2       1.00      1.00      1.00         6
           3       1.00      1.00      1.00         7
           4       1.00      1.00      1.00         5
           5       1.00      1.00      1.00         5
           6       1.00      1.00      1.00         3
           7       1.00      1.00      1.00         3
           8       1.00      1.00      1.00         8
           9       1.00      1.00      1.00         5
          10       1.00      1.00      1.00         8

    accuracy                           1.00        64
   macro avg       1.00      1.00      1.00        64
weighted avg       1.00      1.00      1.00        64
```

Figure 15. Classification Report Skimage

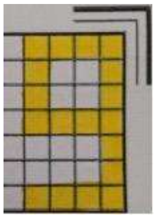
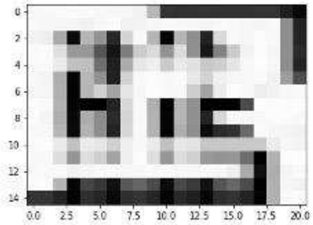
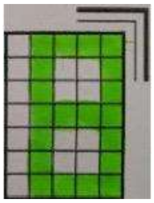
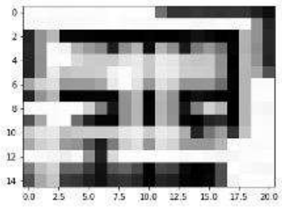
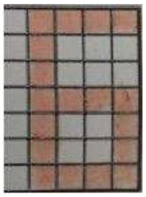
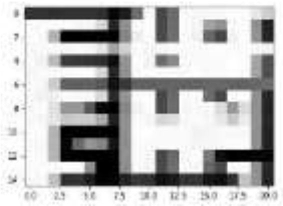


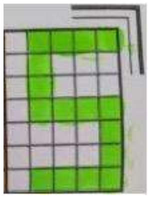
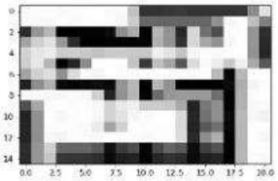
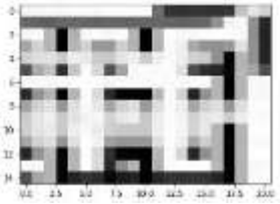
```
Classification report:
              precision    recall  f1-score   support

           0       0.86      1.00      0.92         6
           1       1.00      1.00      1.00         8
           2       0.86      1.00      0.92         6
           3       0.70      1.00      0.82         7
           4       1.00      0.80      0.89         5
           5       0.80      0.80      0.80         5
           6       0.67      0.67      0.67         3
           7       1.00      0.67      0.80         3
           8       1.00      0.75      0.86         8
           9       1.00      0.80      0.89         5
          10       1.00      1.00      1.00         8

    accuracy                           0.89        64
   macro avg       0.90      0.86      0.87        64
weighted avg       0.91      0.89      0.89        64
```

Figure 16. Classification Report GIMP

The data that wasn't accurately detected when seen from the original image was caused by a lot of missing information from the image data, and the data that lost a lot of information when viewed from the original data source was data that had yellow and light green colors.

Table 2. SVM Model Unrecognized Data

Original data	Data after image processing	Data labels
		‘9’
		‘8’
		‘6’

		‘5’
		‘7’

The support vector machine model used in this study can predict data from automatic image processing using a schema with 100% accuracy, while it can only predict data from manual image processing with GIMP 1.20.30 with 89% accuracy. This demonstrates how image processing has a significant impact on how well a model can predict data. The disparity in accuracy between the schematic's automatic and human image processing using GIMP 1.20.30 serves as proof of this. It is preferable to stay away from coloring the data with yellow and light green if you want the model to be accurate. It is clear from hand image processing that these two colors result in a significant loss of information (features) from the data.

The amount of data utilized for training and testing in this study is just one of several variables that influence how well a predictive model performs. When compared to open datasets that are accessible on the internet, which reach 60,000 training data and 10,000 testing data, the amount of data used—only 256 training data and 64 testing data—is still incredibly modest. Additionally, it is preferable to employ dark hues for the same research object to prevent significant amounts of information (features) from being lost during picture processing.

Furthermore, the technique can be tested using the deep learning technique for additional research. This technique is frequently used for picture enhancement and computer vision tasks like face identification. In order to demonstrate that a feature is accurate with the original picture data, it is also advised to display the feature value of an image following the feature extraction procedure.

Conclusion

This study's support vector machine model can predict data from automated image processing using a schema with 100% accuracy, however, it can only do so with 89% accuracy for data from manual image processing using GIMP 1.20.30. This is a nice illustration of how image processing has a significant influence on how accurately a model predicts data. The accuracy difference serves as evidence of this. If we want the model to work better, we need to avoid yellow and light green in the data. Hand image processing shows that there is a substantial loss of information in the data when these two colors are used (features).

References

Arif Muntasa. (2015). *Pengenalan Pola*. Graha Ilmu.

Bai, X., Zhou, F., & Xue, B. (2012). Image enhancement using multi scale image features extracted by top-hat transform. *Optics and Laser Technology*, *44*(2), 328–336. https://doi.org/10.1016/j.optlastec.2011.07.009

Burges, C. J. C., & Schölkopf, B. (1997). Improving the accuracy and speed of support vector machine. *Advances in Neural Information Processing Systems*, 375–381.

Drs. Ulber Silalahi, M. (2014). Metode Dan Metodologi Penelitian. *Bina Budhaya Bandung*, 1999.

Gorgevik, D., & Cakmakov, D. (2001). Handwritten digit recognition by combining SVM classifiers. *EUROCON 2005 - The International Conference on Computer as a Tool*, *II*, 1393–1396. https://doi.org/10.1109/eurcon.2005.1630221

Guerroudji, M. A., & Ameur, Z. (2016). A new approach for the detection of mammary calcifications by using the white Top-Hat transform and thresholding of Otsu. *Optik*, *127*(3), 1251–1259. https://doi.org/10.1016/j.ijleo.2015.10.192

Kowalczyk, A. (2017). Support Vector Machines Succintctly, Syncfusion. *Succinctly E-Book Series*, 114. www.syncfusion.com.

Kumparan News. (2019). *KPU Jelaskan Penyebab Rekapitulasi Suara di Kecamatan Berjalan Lama*. Kumparan.Com.

Kunaver, M., & Tasič, J. F. (2005). Image feature extraction - An overview. *EUROCON 2005 - The International Conference on Computer as a Tool*, *I*, 183–186. https://doi.org/10.1109/eurcon.2005.1629889

Kusmadewi, S. (2003). *Artificial Intelligence (Teknik dan Aplikasinya)* (1st ed.). Graha Ilmu.

Lu, J. (n.d.). *A New Machine Learning Method based*. *20002*, 187–190.

Mauridhi Hery, P., & Arif Muntasa. (2010). *Konsep Pengolahan Citra Digital dan Ektraksi Fitur* (1st ed.). Graha Ilmu.

MB Herlambang. (2019). *Machine Learning: Support Vector Machines*. https://www.megabagus.id/machine-learning-support-vector-machines/

Otsu, N., Smith, P. L., Reid, D. B., Environment, C., Palo, L., Alto, P., & Smith, P. L. (1979). Otsu_1979_otsu_method. *IEEE Transactions on Systems, Man, and Cybernetics*, *C*(1), 62–66.

Polesel, A., Ramponi, G., & Mathews, V. J. (2000). Image enhancement via adaptive unsharp masking. *IEEE Transactions on Image Processing*, *9*(3), 505–510. https://doi.org/10.1109/83.826787

Rizanti, N. A., Arini, & Setianingrum, A. H. (2016). Colon detection using Principal Component Analysis (PCA) and Support Vector Machine (SVM). *Proceedings of 2016 4th International Conference on Cyber and IT Service Management, CITSM 2016*, 1–6. https://doi.org/10.1109/CITSM.2016.7577526

Salcedo-Sanz, S., Rojo-Álvarez, J. L., Martínez-Ramón, M., & Camps-Valls, G. (2014). Support vector machines in engineering: An overview. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *4*(3), 234–267. https://doi.org/10.1002/widm.1125

Tuba, E., Tuba, M., & Simian, D. (2016). Handwritten Digit Recognition by Support Vector Machine Optimized by Bat Algorithm. *Conference on Computer Graphics, Visualization and Computer Vision*, *4617*, 369–376.

Van Der Walt, S., Schönberger, J. L., Nunez-Iglesias, J., Boulogne, F., Warner, J. D., Yager, N.,

Submission: 16 November 2022
Acceptance: 28 November 2022

Gouillart, E., & Yu, T. (2014). Scikit-image: Image processing in python. *PeerJ*, *2014*(1), 1–18. https://doi.org/10.7717/peerj.453

Yodha, J. W., & Kurniawan, A. W. (2014). Perbandingan Penggunaan Deteksi Tepi Dengan Metode Laplace , Sobel Dan Prewit Dan Canny Pada Pengenalan Pola. *Techno.COM*, *13*(3), 189–197.

Zhao, J., Wu, H., & Chen, L. (2017). Road Surface State Recognition Based on SVM Optimization and Image Segmentation Processing. *Journal of Advanced Transportation*, *2017*. https://doi.org/10.1155/2017/6458495

Submission: 16 November 2022
Acceptance: 28 November 2022