Philippine Journal of Science
151 (1): 303-315, February 2022
ISSN 0031 - 7683
Date Received: 07 Sep 2021

Block-level Optical Character Recognition System for Automatic Transliterations of *Baybayin* Texts Using Support Vector Machine

Rodney B. Pino[1,2], **Renier G. Mendoza**[1]*, and **Rachelle R. Sambayan**[1]

[1]Institute of Mathematics, University of the Philippines Diliman
Quezon City, Metro Manila, Philippines
[2]Department of Mathematics and Computer Science
University of the Philippines Baguio, Governor Pack Road 2600 Baguio, Philippines

Baybayin **is a Tagalog-language writing system primarily used in the northern Philippines during the pre-Hispanic period. In 2018, the House of Representatives approved House Bill 1022 or the "National Writing System Act," which declares the *Baybayin* script as the Philippines' national writing system. Thus, documents, signages, books, *etc.* may soon have *Baybayin* texts. However, the Latin alphabet is still the primary script used in the country. Hence, it is possible that Latin and *Baybayin* scripts may be found on the same text. In this paper, we present an optical character recognition (OCR) system that identifies *Baybayin* scripts from Latin in a text image. The preprocessing method applies the conversion of the input image to binary data and calculating the respective bounding box of each word found from the text, where we utilize a modified $k-$ means algorithm and MATLAB ocr function, respectively. The classification then involves isolating each word and further segmenting each character's components. With the aid of a support vector machine (SVM) character classifier, we determine the word's script by the highest number of characters classified into either *Baybayin* or Latin. To the best of our knowledge, this is the first system that discriminates, at the block level, the *Baybayin* script from Latin. The proposed algorithm yields a 93.64% recognition accuracy tested in a novel dataset. The accompanying code of the proposed algorithm and the dataset are made publicly available to make the results of the study reproducible.**

Keywords: *Baybayin* and Latin word script identification, *Baybayin* word transliteration, support vector machine, optical character recognition

INTRODUCTION

Baybayin is one of the pre-colonial writing systems used, particularly by the Tagalog people in the Philippines (Cabuay 2009). Evidence shows how *Baybayin* was an integral part of the way of life of Filipinos during the pre-Hispanic period (Lagunsad 2020). Many cultural advocates, organizations, and working bodies are making efforts to preserve, reintroduce, and propose ways to show the significance of *Baybayin* for the identity of the Philippine nation. Furthermore, the *Baybayin* writing system contributes to the cultural identity and socio-psychological well-being of *Baybayin* advocates and scholars (Camba 2021). In April 2018, the Committee on Basic Education Culture of the Philippine Congress signed House Bill 1022 or the "National Writing System Act," which declares the *Baybayin* script as the Philippines'

*Corresponding Author: rmendoza@math.upd.edu.ph

national writing system. The said Bill mandates all local government units to inscribe *Baybayin* with its Latin translation in their communication systems (*e.g.* signages and public documents). Further, local manufacturers are required to imprint *Baybayin* scripts with their translation on product labels, and at least four executive departments are tasked to promulgate the *Baybayin* writing system (Lim and Manipon 2019).

A Tagalog-based language, the *Baybayin* writing system is an abugida or alpha syllabary (see Figure 1 for the comparison between the Latin alphabet and the *Baybayin* script). It consists, traditionally, of 17 distinct characters: 14 (syllabic) consonants and three vowels (Figure 1B). Each consonant character is pronounced with an inherent vowel sound `\a\', and diacritics (usually a dot or a bar) are used to express the other vowels. For instance, a diacritic written above a consonant character may represent an accompaniment `\e\' or '\i\', while a diacritic written below may represent an `\o\' or `\u\' sound. One can also use diacritics (usually a cross or X symbol) to silence the vowel sound of a consonant character (Cabuay 2009). Figure 2 shows an example of a phonetic distinction in the *Baybayin* character using diacritical marks.

OCR is the classification of optical mechanisms in a digital image. It is a technology that allows the conversion of different types of documents such as PDF (portable document format) files, scanned paper documents, or images captured by a digital camera into searchable and editable data (Chaudhuri *et al.* 2016). There is significant progress in OCR research studies for the past decade. One aspect is on discriminating the scripts of different writing systems. Determining the script before choosing an appropriate character recognition algorithm is important for the effectiveness

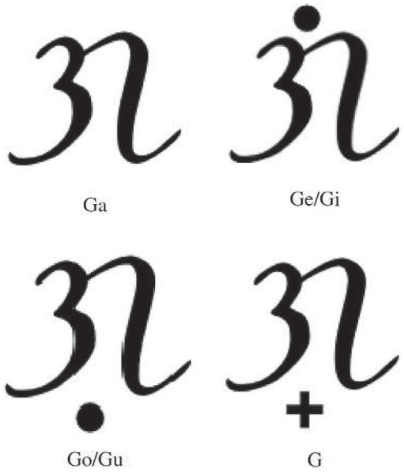

Figure 2. Incorporating diacritics in a *Baybayin* consonant character.

of the OCR system. Most studies in script recognition consider either printed or handwritten scripts and are based on several levels: page, line, word, and character level (Ghosh *et al.* 2010). Recent OCR systems using different machine learning algorithms for various scripts can be found in the literature (Heo *et al.* 2021a, b; Chumwatana and Rattana-umnuaychai 2021; Gunawan 2017; Septiana 2021; Horie *et al.* 2021; Singh *et al.* 2015; Rajput and Ummapure 2017; Awel and Abidi 2019; Memon *et al.* 2020; Bhunia *et al.* 2019).

Research works on *Baybayin* OCR only started to gain popularity recently. The first *Baybayin* OCR system was done by Recario *et al.* (2011), where they have explored the use of an automated reader for *Baybayin* characters that outputs the corresponding Latin syllable. Their study uses the freeman chain coding and line angle categorization, where they acquired 51.96%

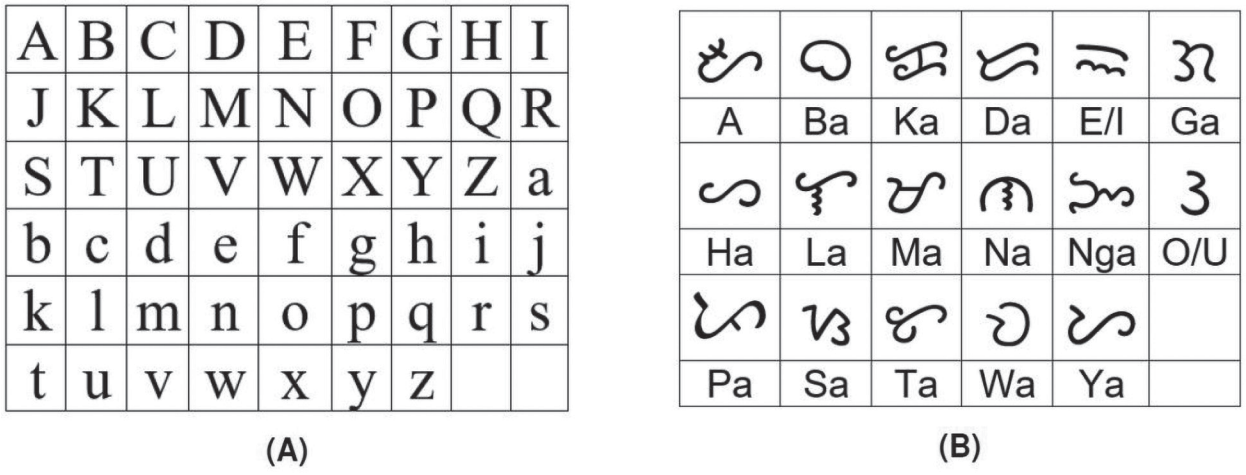

Figure 1. (A) Latin alphabet in upper-and lowercase; (B) *Baybayin* characters with equivalent Latin transliteration.

and 66.47%, respectively. In the study of Nogra *et al.* (2019b, 2020), *Baybayin* character recognition schemes that output the transliterated syllable in Latin of the input *Baybayin* character have been presented. They have utilized the convolutional neural network (CNN) (Nogra *et al.* 2020) and LSTM neural network (Nogra *et al.* 2019b) models and yielded a result of 94 and 92.9% recognition rates, respectively. The use of the inception network improved the average validation accuracy to 96.2% (Nogra 2020). Mobile e-Learning applications have been developed based on these models (Nogra *et al.* 2019a; Nogra 2020). With CNN and feed-forward neural network models, Daday *et al.* (2020) have proposed a *Baybayin* script recognition that converts the character into its equivalent Latin unit/syllable. The two model types used a dropout method and obtained 91.69% and 92.4% recognition rates, respectively. Bague *et al.* (2020) have presented a CNN model for *Baybayin* recognition with a Visual Geometry Group 16 (VGG16 type network) architecture, where they acquire a 98.84% reading accuracy. These *Baybayin* OCR studies are based momentarily on the character level, which signifies its early development. Moreover, Recio and Mendoza (2019) employed an edge detection approach to recognizing text images containing *Baybayin* characters.

In the study of Pino *et al.* (2021b), it has been shown how SVM is very effective in discriminating *Baybayin* scripts from the Latin alphabet. The SVM model obtained a 98.5% recognition rate. SVM is a supervised machine learning algorithm that attracted researchers in data classification due to its high recognition accuracy and robustness (Thomé 2012). This machine-learning algorithm has been used extensively in various applications (Thomé (2012); Nayak *et al.* (2015); Rivero *et al.* (2017); Rivero and Kato (2018); Do and Le (2019)). The *Baybayin* script OCR using SVM by Pino *et al.* (2021b) has been extended to *Baybayin word level* in the other study (Pino *et al.* 2021a). This is the first OCR system that can classify *Baybayin* at the word level, with a recognition accuracy of 97.9%.

In this study, we extend the results in the studies of Pino *et al.* (2021a, b) to block level, which may contain several lines of text. This means that the input image may contain multiple *Baybayin* and/or Latin words. This is again the first OCR study that can classify *Baybayin* at the block level. The system presented here detects the number of words in the input text image and further segments each word to its character components for script classification. The word's script category depends on the higher number of characters that belong to the corresponding script. If the system detects a *Baybayin* word/s in the input text image, it proceeds with the transliteration of each detected *Baybayin* word. Our proposed system has two image outputs:

1. script classification of each word detected by the system, either *Baybayin* or Latin; and
2. equivalent Latin transliteration of *Baybayin* words detected from the input text image.

The remainder of the paper proceeds as follows: "Materials and Methods" section discusses how the *Baybayin* and Latin text images are collected and how the proposed block-level *Baybayin* OCR works; "Results and Discussion" and "Conclusion" contain the discussion of the results and the conclusion and future works, respectively.

MATERIALS AND METHODS

Dataset Collection

Data preparation is an essential part of every recognition system. In this section, we present how we gather the images of *Baybayin* and/or Latin (block) text images. The dataset used in this study can be accessed publicly in the repository made by Pino (2021b).

We collected most of these images from different websites. Text images of either hand- or typewritten *Baybayin*, Latin, or *Baybayin* and Latin writings were saved using a snipping tool. Figure 3 shows a preview of these images.

A total of 110 text images were gathered. The specification of the dataset is presented in Table 1. We used more block images that contain the combinations of *Baybayin* and Latin texts (70 out of 110) to test how the method discriminates *Baybayin* from Latin scripts in an image block.

The Proposed System

In the study of Pino *et al.* (2021a), the *Baybayin* OCR at the word level assumes that the input of the system is already in *Baybayin*. The system identifies the Latin equivalent of the already-assumed-*Baybayin* input word. However, this system does not discriminate *Baybayin* from Latin words. Because Filipinos primarily use Latin in writing, *Baybayin* and Latin words may be written on the same block. In this work, we present an algorithm that discriminates all the *Baybayin* words from Latin words on a text block. After identifying all the *Baybayin* words on the page, we use the scheme proposed in the study of Pino *et al.* (2021a) to find the Latin transliterations of each *Baybayin* word. The system identifies all the words in the text image. Each word is then segmented into its character components, which are then categorized into either Latin or *Baybayin*. We used MATLAB (vR2020a) to code and

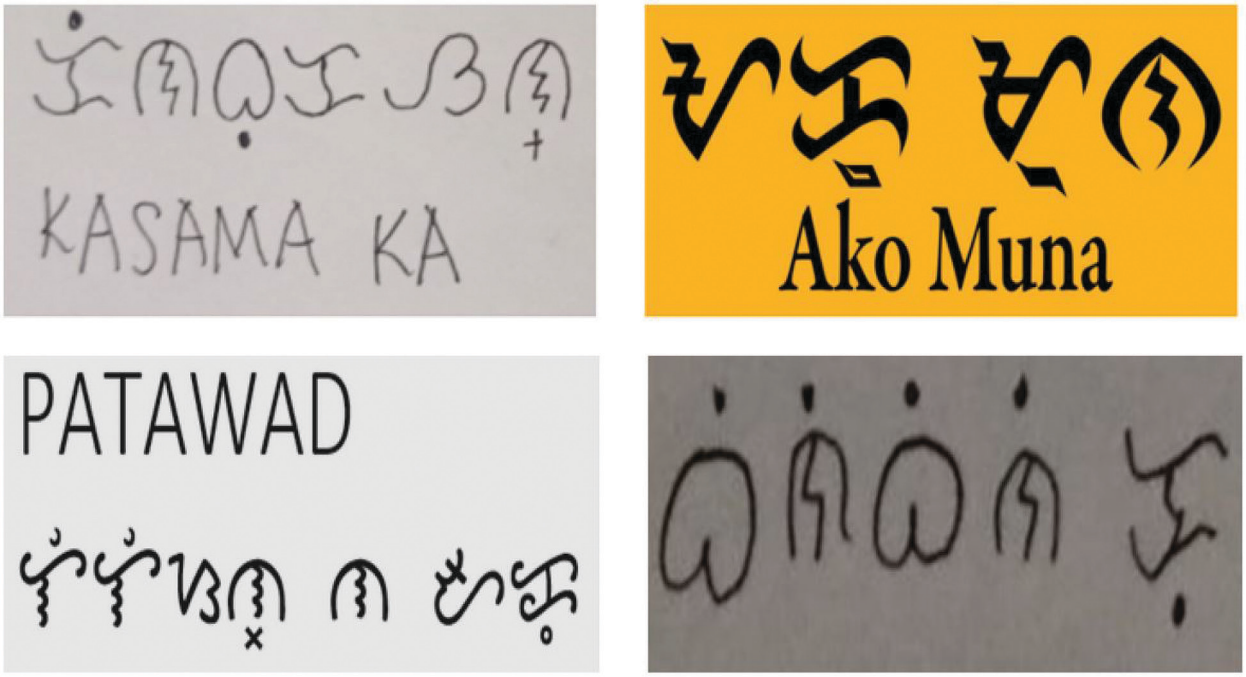

Figure 3. Samples of *Baybayin* and Latin text images in the dataset collection.

Table 1. Number of images that have Baybayin and/or Latin inscriptions written by hand or typed.

Text specifications	Number of images
Hand-written *Baybayin*	10
Hand-written Latin	10
Type-written *Baybayin*	10
Type-written Latin	10
Hand-written *Baybayin* and Latin	35
Type-written *Baybayin* and Latin	35

implement the recognition system. Our proposed method uses the *Baybayin* OCR for word recognition proposed in the study of Pino *et al.* (2021a), which relies heavily on the OCR for *Baybayin* characters presented in the study of Pino *et al.* (2021b). Hence, each *Baybayin* character on the input image must satisfy all the assumptions set in the study of Pino *et al.* (2021a, b), and are presented as follows:

- the text print is darker than the background;
- the main body of the character is larger than its diacritic; and
- the diacritic is not touching the main character, written above or below its respective main character and is within the width of the main character.

The following assumptions are needed to guarantee that the characters are properly extracted, the number of words is correctly counted, and a word is fairly categorized into its correct script classification:

- all characters (including diacritics) in the word are not attached to other characters;
- there is a bearable space between each word; and
- all characters in a word must belong to one script only.

The proposed system starts by converting the input image to binary data. A modified $k-$ means function is used for this operation, where lower intensities are converted into 1s (white pixels) and high intensities are changed to 0s (black pixels). We then implement the MATLAB built-in ocr function and utilize the text properties: word count (N) and word bounding boxes (WBB). The ocr function, which is set to text block recognition layout, reads in a left-to-right, top-to-bottom manner. Depending on the number of words found, the system then categorizes each word into either *Baybayin* or Latin script.

We define the input text image T as the set of words $\{W(p)\}_{p=1}^{N}$, where p denotes the order of the word and N is the number of words counted by the ocr function. Each of $W(p)$ is then extracted using their respective WBB. The MATLAB regionprops function is then implemented to provide the following measurements for each character

from the extracted word $W(p)$: area, bounding box, and centroid. We further define $W(p)$ as the set of characters $\{char(i)\}_{i=1}^{M}$, where i stands for the sequence of the character and M as the number of characters in $W(p)$.

A quick approach to determine whether a word is *Baybayin* or Latin is by finding the script classification of the first character of the word. However, since the script classification presented in the study of Pino *et al.* (2021b) is 98.5%, there is a small chance that the first character might be incorrectly classified. Thus, given a word image $W(p)$, we classify all its character components into their script categories. If in a given word, the number of recognized *Baybayin* characters is more than the number of recognized Latin characters, the word is classified as *Baybayin*. Otherwise, the word is classified as Latin. Starting with $p = 1$, we set $L = 0$ (for Latin) and $B = 0$ (for *Baybayin*). The procedure follows with isolating each $char\ (i)$. If a diacritic exists (for *Baybayin*), the character's main body C is further extracted; otherwise, $char\ (i) = C$ is assumed. The feature extraction utilized in the study of Pino *et al.* (2021b) is then implemented to obtain the $1 \times 3{,}136$ feature vector. The resulting feature vector will be fed to the *Baybayin* and Latin SVM script classifier. If C is classified as *Baybayin*, -1 is added to B, else, $+1$ is added to L. The process is repeated to each $char\ (i)$. $W(p)$ is classified as *Baybayin* if $sign\ (L + B)$ is negative whereas Latin if otherwise. The procedure is repeated iteratively for all $p \in \{2,3, \dots, N\}$. The system then outputs an image, where the *Baybayin* and Latin texts are boxed with different colors (red for *Baybayin* and blue for Latin). We denote by $W(p)_B$ a word that is classified as *Baybayin*.

All $W(p)_B$'s will be fed into the *Baybayin* Word Recognition System ($BWRS$) presented in the study of Pino *et al.* (2021a). $BWRS$ returns all the equivalent Latin forms of the input *Baybayin* word $W(p)_B$. After extracting the transliterations of every $W(p)_B$, it is followed by replacing the *Baybayin* inscriptions of T with their respective transliterations. If $W(p)_B$ has multiple Latin equivalents, the system caters to it by combining every generated Latin word to other Latin word inscriptions. Otherwise, the system displays only the first image output.

Figure 4 shows a visual presentation of the complete process. Starting from the input image (see Figure 4A), the system converts the image to binary data as shown in Figure 4B. Using MATLAB ocr function, the system then determines its detected words bounding region as presented in Figure 4C. For the given image in Figure 4A, three words are identified and are denoted by $T = \{W(p)\}_{p=1}^{3}$. Figure 4E shows the script classification of each word given in 4D. The character script categorization procedure in the study of Pino *et al.* (2021b) is employed

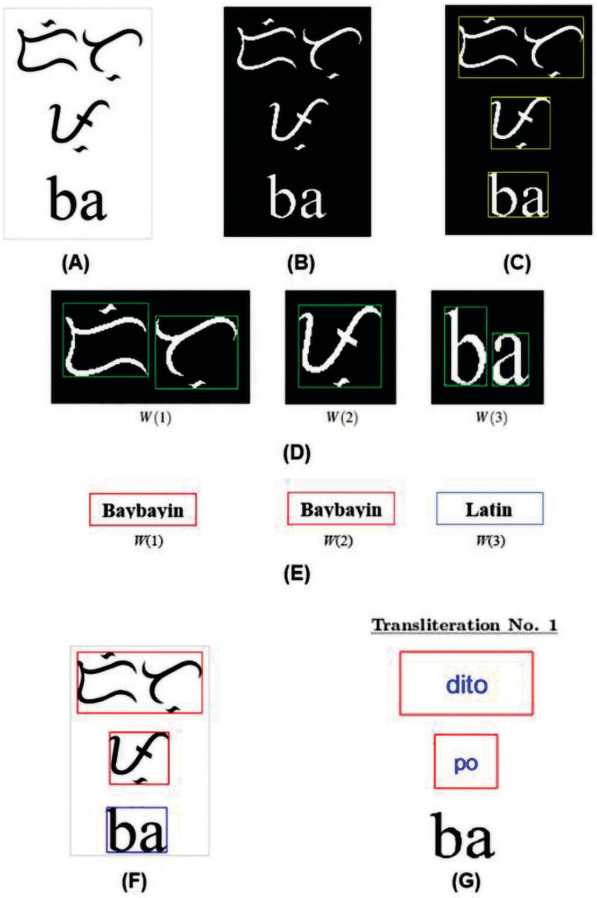

Figure 4. The system process: (A) raw text image; (B) binary conversion; (C) words extraction; (D) segmentation of $W(p)$ into its M character components; (E–F) script identification; (G) *Baybayin* to Latin transliteration.

in each character. We used this classification scheme because of its impressive recognition rates in terms of accuracy, precision, recall, and F1 score (see Table 2). For the detailed discussion of this method, we refer the readers to the study of Pino *et al.* (2021b).

The word's script category depends on the higher number of Latin or *Baybayin* characters classified. The image in 4F shows the classification of the *Baybayin* and Latin scripts in the text. *Baybayin* words are boxed in red, while the word in the Latin alphabet is boxed in blue. A possible transliteration of the *Baybayin*-classified words is shown in Figure 4G, where we make use of the system introduced in the study of Pino *et al.* (2021a).

The *Baybayin* block recognition system is set only to generate at most three transliterations of each *Baybayin*-classified word found and to present at most six text transliterations of the input image.

Figure 5 shows the overall flow of the proposed algorithm. Algorithms 1 and 2 show the procedure of word script

Table 2. Recognition rates of the utilized SVM models presented by Pino *et al.* (2021b).

SVM character classifiers	Performance measures			
	Accuracy	Precision	Recall	F1 score
Baybayin and Latin scripts	98.56	98.56	98.55	98.55
Baybayin characters	96.20	96.20	96.21	96.19
Baybayin diacritics	100.00	100.00	100.00	100.00

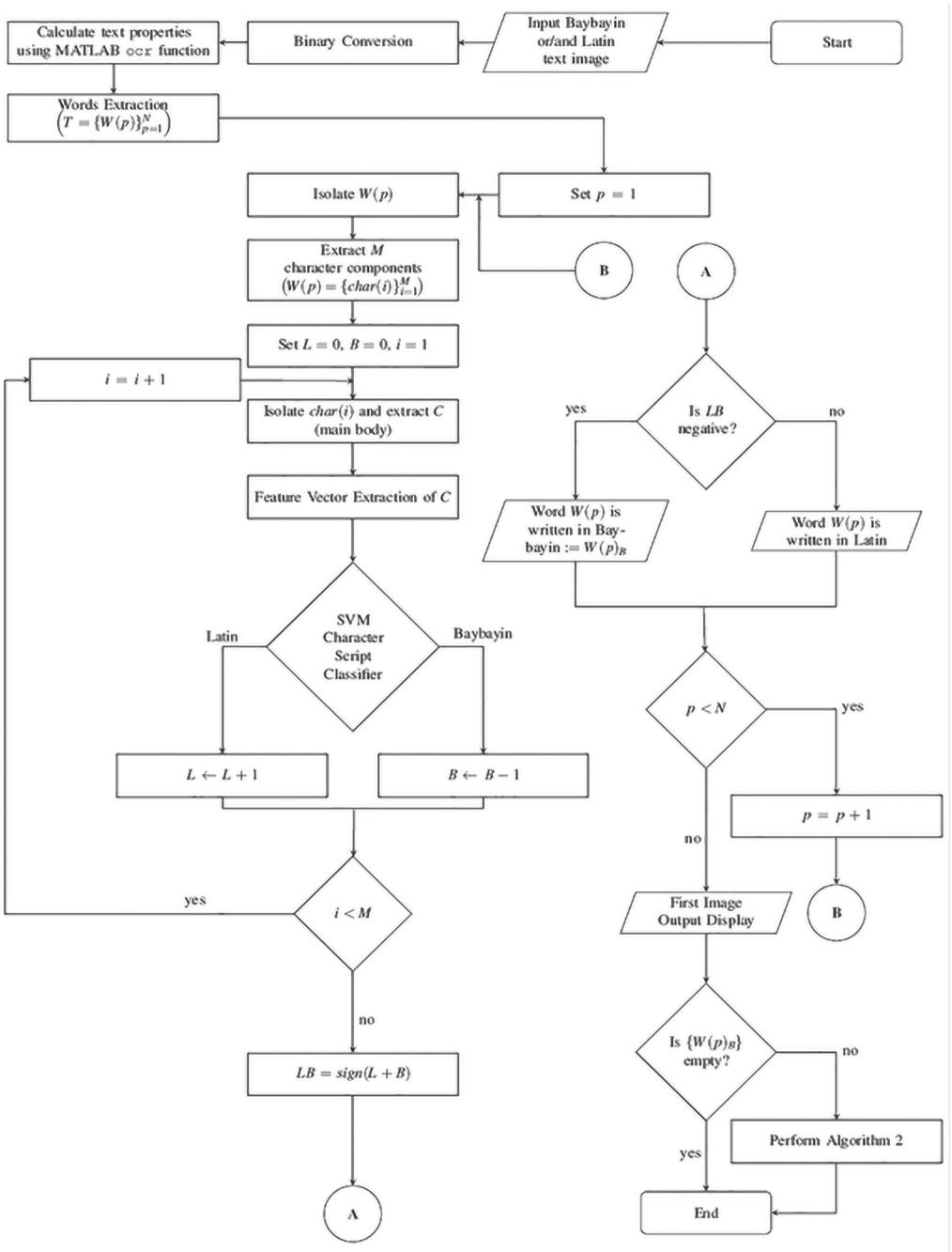

Figure 5. Script identifier flowchart.

Algorithm 1. Script discriminator.

Require: *Baybayin* and/or Latin text image.
Ensure: Output image with each word boxed by their respective script color.

Convert the input image into binary data.
Calculate text properties using MATLAB ocr function.
Extract each word found from $T = \{W(p)\}_{p=1}^{N}$.
for $p = 1{:}N$ **do**
  Isolate $W(p)$
  Extract M character components to form $W(p) = \{char(i)\}_{i=1}^{M}$.
  Set $L = 0$ and $B = 0$.
  for $i = 1{:}\ M$ **do**
    Let $C = char\ (i)$ main body. Isolate C from $char\ (i)$.
    Feature vector extraction of C.
    Feed the resulting feature vector to SVM Script Classifier.
    If $Baybayin$ **then**
      $B \leftarrow B - 1$
    else
      $L \leftarrow L - 1$
    end if
  end for
  if $sign\ (L + B)$ is negative **then**
    Word $W(p)$ is written in *Baybayin* script $\coloneqq W(p)_B$.
  else
    Word $W(p)$ is written in Latin script.
  end if
end for
Display first output image.
if $\{W(p)_B\} \neq \emptyset$ **then**
  Feed input image and $\{W(p)_B\}$ to Algorithm 2.
else
  Return.
end if

Algorithm 2. *Baybayin* inscriptions transliterator.

Require: *Baybayin* and/or Latin text image and $\{W(p)_B\}$.
Ensure: Output image where each *Baybayin*-classified word is replaced by their respective Latin equivalent.

for $p = 1{:}\ N$ **do**
  if p satisfies $W(p)_B$ **then**
    Feed $W(p)_B$ to $BWRS$-.
    $U =$ Latin transliteration/s of the *Baybayin* word $W(p)_B$.
    Replace $W(p)_B$ with U respecting its WBB.
  else
    Discard.
  end if
end for
Combine all results to the input image.
Display second output image.

classification and the *Baybayin* word transliteration process, respectively. The MATLAB codes used in this study can be accessed publicly in the repository made by Pino (2021a).

RESULTS AND DISCUSSION

The proposed system is applied to 110 *Baybayin* and Latin (block) text images. These images satisfy the system's assumptions stated in the previous section. A test is successful if all words in the image are correctly categorized into their script. After implementing the proposed system to the dataset, 103 images (out of 110) were correctly classified to their corresponding script. That is, all detected *Baybayin* and Latin words in the 103 images are boxed with their corresponding script color in the output image.

The seven misclassified text images occurred in two Latin handwritten images and five handwritten images with Baybayin and Latin inscriptions. Table 3 shows the breakdown of their misclassification. There are only seven misclassified words among all the combined words in the dataset of text blocks.

Sample results of correctly classified text block images and Baybayin transliterated words are shown in Figures 6, 7, and 8. Words bounded by red boxes are Baybayin, while the blue bounded ones are script written in Latin. Aside from the image with the color-coded bounding boxes, the system will also display the number of recognized Baybayin and Latin words. For instance, for the block image in Figure 6A, the created MATLAB code will display the text:

```
'Total number of words found is 4. It consists of 3 Baybayin word/s (bounded by red boxes) and 1 Latin word/s (bounded by blue boxes).'
```

In addition, since the system detects that there are *Baybayin* words in the input text image, it further transliterates each *Baybayin* word into the Latin alphabet and replaces the *Baybayin* inscriptions, as shown in Figure 6B.

In Figure 7, the system successfully identified the *Baybayin* words from the Latin ones. For this example, one of the *Baybayin*-classified words has multiple Latin transliterations. The algorithm proceeds by allocating each equivalent transliteration in one image and combining it with the other Latin inscriptions. Figure 7B shows this allocation for multiple transliterations. The words *bobo* and *bubo* are two distinct Tagalog words that mean "dumb" and "to pour" in English, respectively.

Figure 8B shows the image outputs for two or more *Baybayin*-classified words with multiple transliterations. The system takes each *Baybayin* word with multiple Latin equivalents and combines it with the other *Baybayin*

Table 3. Breakdown of misclassified (misc.) images.

Text specifications	No. of words (Latin/ *Baybayin*)	Misc. Latin/*Baybayin* (in words)
Hand-written Latin 1	18 / 0	1 / 0
Hand-written Latin 2	9 / 0	1 / 0
Hand-written *Baybayin* and Latin 1	3 / 3	0 / 1
Hand-written *Baybayin* and Latin 2	3 / 3	0 / 1
Hand-written *Baybayin* and Latin 3	2 / 1	0 / 1
Hand-written *Baybayin* and Latin 4	1 / 2	0 / 1
Hand-written *Baybayin* and Latin 5	2 / 3	0 / 1

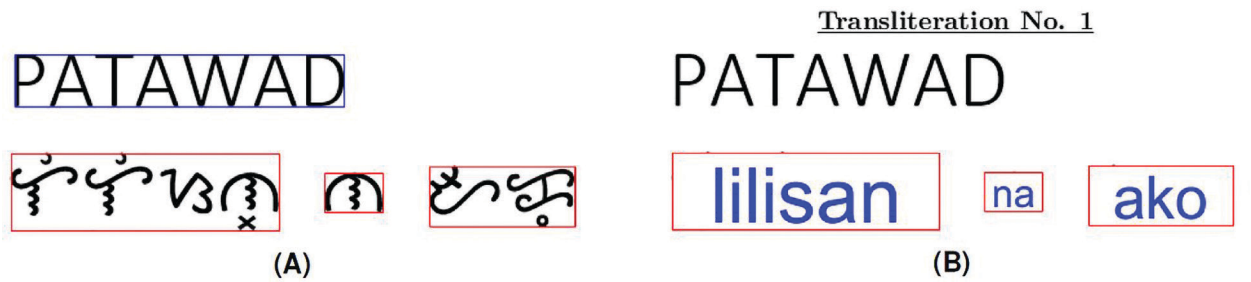

Figure 6. Sample results of the algorithm applied to a text image: (A) the system correctly distinguished the Latin from Baybayin words (red for Baybayin and blue for Latin) and (B) correctly transliterate all the detected Baybayin words.

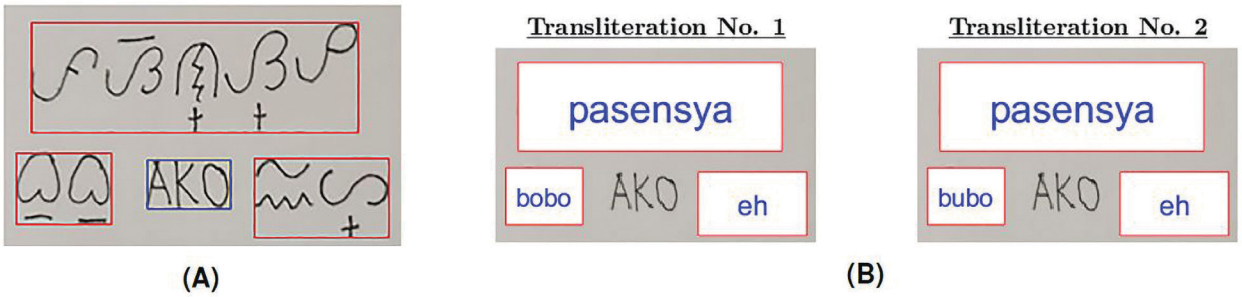

Figure 7. Sample text image with one Baybayin word with multiple Latin equivalents: (A) the system successfully identified Baybayin words from the Latin writings through the color-coded bounded boxes, and (B) determined the multiple Latin transliterations.

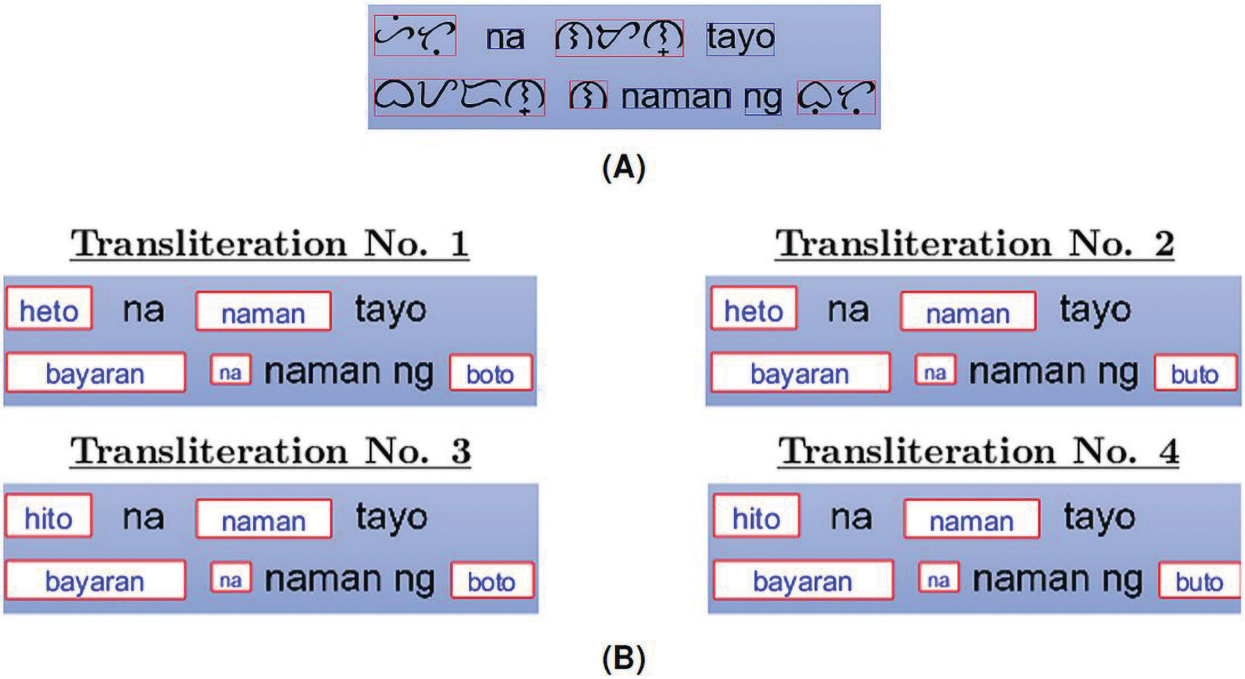

Figure 8. Sample text image with two Baybayin words that have multiple Latin transliterations: (A) correct word script classification of each detected word and (B) corresponding multiple Latin transliterations.

word with multiple corresponding Latin expressions. Afterward, each unique combination is merged with the other Latin inscriptions. The Tagalog words *heto* and *hito* have different meanings. In English, they mean "here" and "freshwater catfish," respectively. Also, *boto* means "vote" and *buto* means "bone" in the English language.

Although the system works well in general, there are cases when it is less successful. In Figure 9, the system did well in discriminating Latin from *Baybayin* scripts, but there were three instances (among the 43 identified text blocks) where misclassifications happened. For example, the first *Baybayin* misclassification (uppermost right) was due to the bar '|' symbols, which may have been recognized as the Latin small letter "l" or the big Latin letter "I." The letter "D" was also misclassified because the first assumption of the proposal was not satisfied. The image in Figure 9 was not included in the dataset because it does not satisfy the system's assumptions. We added this example to illustrate that even though the input image is degraded and old, the system can still be effective.

The transliteration process relies on the correct classification of each character in the text block. A word might be transliterated incorrectly if one of the characters in a word is misclassified. This is illustrated in Figure 10 where one of the words was misclassified. The *Baybayin* word "di" was classified Latin by the system (see Figure 10B). Observe that the character is written similarly as the Latin letter "c." Moreover, one of the *Baybayin* words was also transliterated incorrectly (bottom right of the two images in Figure 10C). The second character in this word was misclassified as "a." This happens because the character classification OCR used does not have a 100%

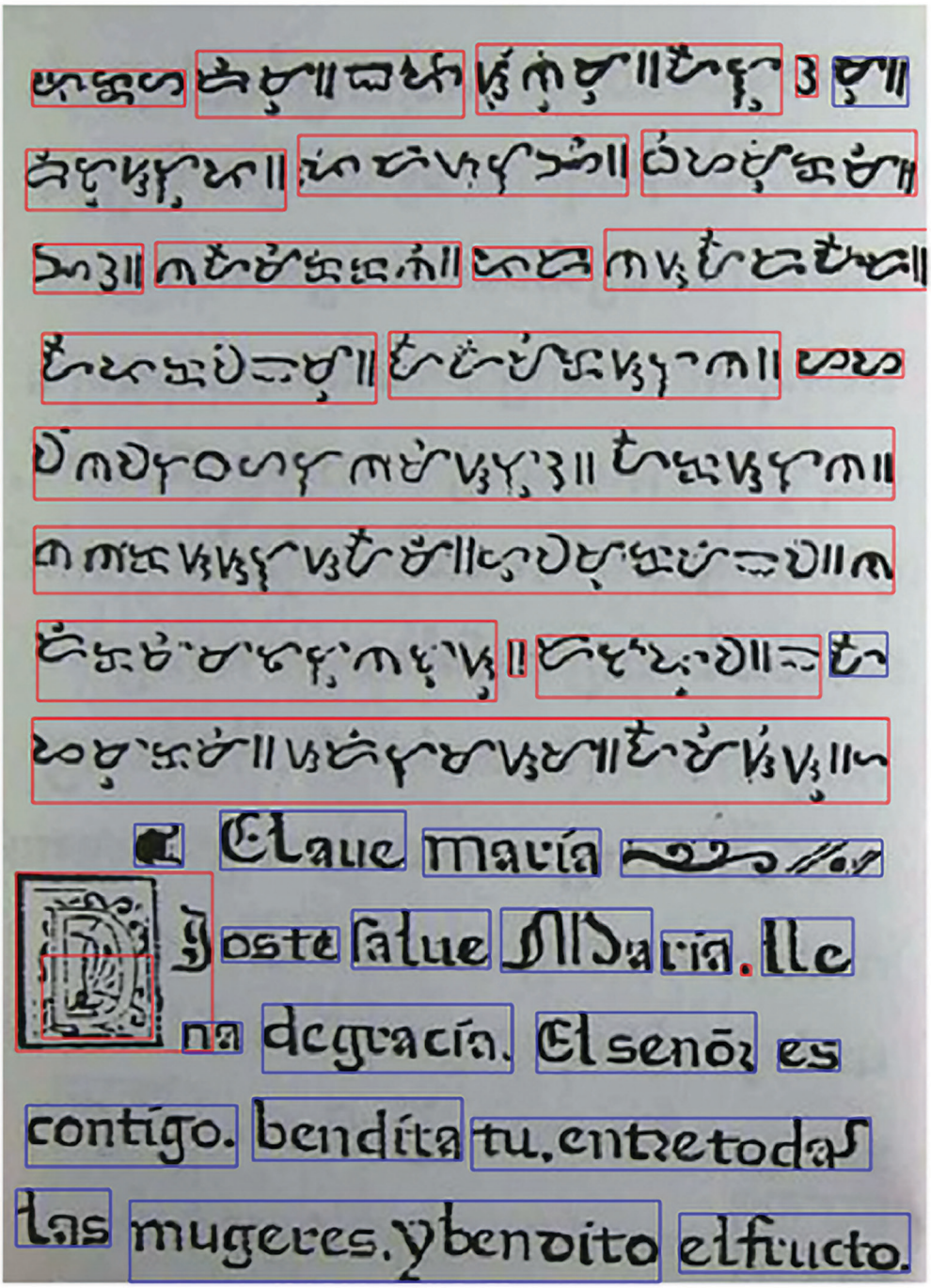

Figure 9. A page of the first Christian book in the Philippines, Doctrina Christiana (Plasencia 1593), fed into the system. This image is taken from Gubernatoria (2010). The words inside the red bounding boxes are recognized by the proposed system as Baybayin and those inside blue bounding boxes as Latin.

recognition accuracy.

The potential cause of errors in the proposed system is summarized as follows:

- the input does not satisfy the system's assumptions,
- misclassification in recognizing the character's script and/or its identification, and
- Latin transliteration of the *Baybayin* word found in the text is not in the Tagalog dictionary.

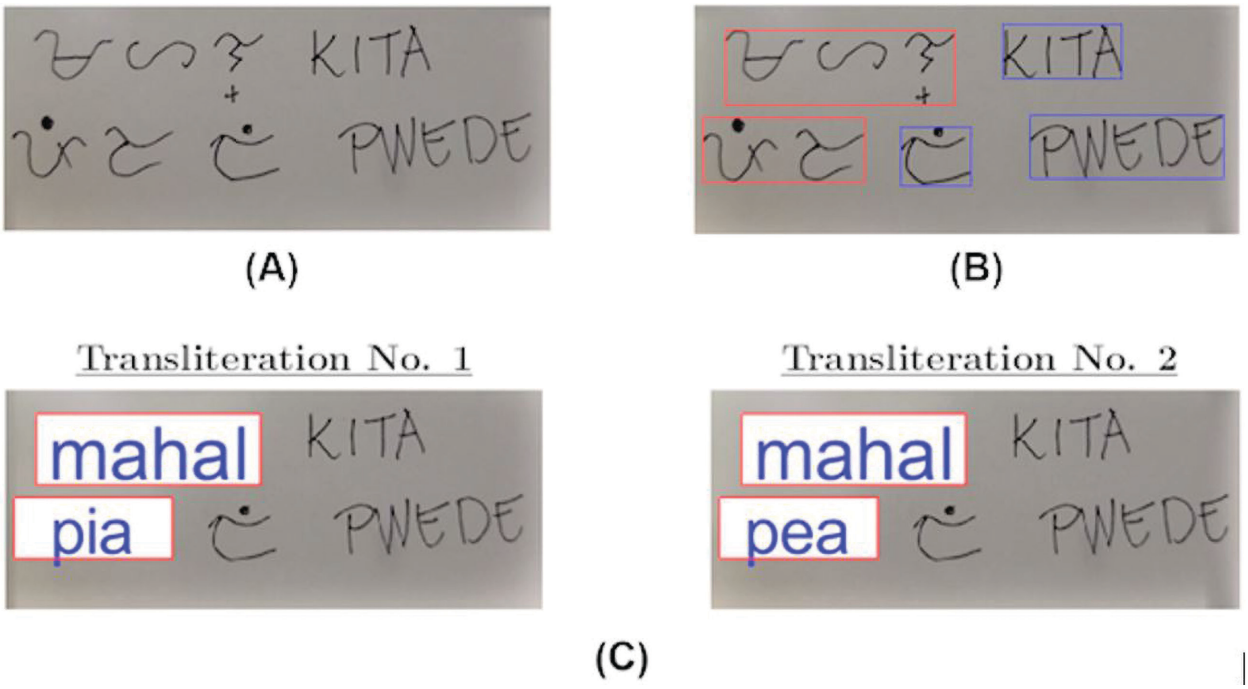

Figure 10. A sample block image with transliterations not in the Tagalog dictionary.

CONCLUSION

This study provides a dataset of images containing both *Baybayin* and Latin words. The dataset has been made available for other researchers interested in studying the computer vision of *Baybayin* scripts. Our work proposes a system that distinguishes *Baybayin* from Latin words on a text image. The gathered dataset and the MATLAB program utilized in this paper can be accessed publicly in the repositories made by Pino (2021b and 2021a, respectively).

The algorithms presented in the studies of Pino *et al.* (2021a, b) are *Baybayin* OCR systems on character and word level, respectively. This work successfully extended these results to the block level. The formulated system shows high recognition accuracy under certain assumptions. Although these assumptions are not restrictive, one can study how the system can be modified for more general use. We relied heavily on SVM in this work, but other researchers can look at other machine learning techniques. As illustrated in Figures 9 and 10, the system can still be improved if the character classification algorithm has a higher recognition accuracy. A comparative study of various machine learning algorithms for *Baybayin* OCR at the block level is a direct consequence of this research.

Note that words written in *Baybayin* are oftentimes ambiguous. For example, *Baybayin* does not distinguish the vowel "o" from "u" (Kabuay 2013; Morrow 2010), which tells that "go" and "gu" are written in the same manner (see Figure 2). Our system can only present all the possible transliterations and does not have the capacity to identify which among the outputs makes sense. For example, in Figure 8, the first two transliterations are both grammatically correct. However, the first transliteration is the more fitting choice. Identifying the correct phrase/s from various transliteration choices needs delving into the syntax of the Tagalog language. This is an exciting future research direction and demands a thorough study.

This work is envisioned to help in promoting *Baybayin* and inspire researchers to pursue studies on computer vision for *Baybayin*.

ACKNOWLEDGMENTS

This work was funded by the University of the Philippines System Enhanced Creative Work and Research Grant (ECWRG-2019-2-11-R).

REFERENCES

AWEL MA, ABIDI AI. 2019. Review on optical character recognition. International Research Journal of Engineering and Technology 6(6): 3666–3669.

BAGUE L, JORDA RJ, FORTALEZA B, EVANCULLA AD, PAEZ MA, VELASCO J. 2020. Recognition of

baybayin (ancient Philippine character) handwritten letters using vgg16 deep convolutional neural network model. International Journal of Emerging Trends in Engineering Research 8(9): 5233–5237.

BHUNIA AK, KONWER A, BHUNIA AK, BHOWMICK A, ROY PP, PAL U. 2019. Script identification in natural scene image and video frames using an attention based convolutional-lstm network. Pattern Recognition 85: 172–184.

CABUAY C. 2009. An Introduction to *Baybayin*. Raleigh, NC: Lulu Press, Inc.

CAMBA AT. 2021. *Baybayin*: the Role of a Written Language in the Cultural Identity and Socio-Psychological Well-being of Filipinos (Doctoral Dissertation). Harvard University.

CHAUDHURI A, MANDIVAYA K, BADELIA P, GHOSH S. 2016. Optical Character Recognition Systems for Different Languages with Soft Computing, 1st ed. Cham, Switzerland: Springer Publishing Company, Incorporated.

CHUMWATANA T, RATTANA-UMNUAYCHAI W. 2021. Using OCR Framework and Information Extraction for Thai Documents Digitization. In: 2021 9th International Electrical Engineering Congress (iEECON). p. 440–443.

DADAY MJ, FAJARDO A, MEDINA R. 2020. Recognition of *baybayin* symbols (ancient pre-colonial Philippine writing system) using image processing. International Journal of Advanced Trends in Computer Science and Engineering 9: 594–598.

DO DT, LE NQK. 2019. A sequence-based approach for identifying recombination spots in *Saccharomyces cerevisiae* by using hyper-parameter optimization in fasttext and support vector machine. Chemometrics and Intelligent Laboratory Systems 194: 103855.

GHOSH D, DUBE T, SHIVAPRASAD A. 2010. Script recognition's review. IEEE Transactions on Pattern Analysis and Machine Intelligence 32(12): 2142–2161.

GUBERNATORIA. 2010. File:doctrinachristianaespanolaytagala8-9.jpg. Retrieved on 17 Apr 2021 from https://tinyurl.com/kn2m2djx

GUNAWAN D, ARISANDI D, GINTING FM, RAHMAT RF, AMALIA A. 2017. Russian character recognition using self-organizing map. Journal of Physics: Conference Series 801(1): 012040.

HEO JH, LEE SW, LEE HW. 2021a. A Comparative Study on the Perception Performance of Handwriting in Korean and English Using Machine Learning. In: 2021 21st ACIS International Winter Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD-Winter). p. 274–275.

HEO JH, LEE SW, LEE HW, GIM GY. 2021b. A Study on the Recognition of Hangeul through Transitional Learning in Handwritten Application. In: International Conference on Intelligence Science. p. 77–89.

HORIE F, GOTO H, SUGANUMA T. 2021. Synthetic Scene Character Generator and Ensemble Scheme with the Random Image Feature Method for Japanese and Chinese Scene Character Recognition. IEICE Transactions on Information and Systems 104(11): 2002–2010.

KABUAY K. 2013. Modified *baybayin*. Retrieved on 21 Jul 2021 from https://blog.kabuay.com/tutorials/modified/

LAGUNSAD RIL. 2020. *Sipat-suri sa mga Katutubong Sulat*: *Salalayan sa Pagbuo ng Mungkahing Manwal sa Baybaying Filipino*. Enderun Colleges Scholarly Review 3(2).

LIM MK, MANIPON RH eds. 2019. *Bilangan* 2: Selected Papers from the 2019 International Conference on Cultural Statistics and Creative Economy. NCCA, Intramuros, Manila, Philippines.

MEMON J, SAMI M, KHAN RA. 2020. Handwritten optical character recognition (ocr): a comprehensive systematic literature review (slr). IEEE Access 8: 142642–142668.

MORROW P. 2010. *Baybayin* – the ancient script of the Philippines. Retrieved on 21 Jul 2021 from http://paulmorrow.ca/bayeng1.htm

NAYAK J, NAIK B, BEHERA HS. 2015. A comprehensive survey on support vector machine in data mining tasks: applications & challenges. International Journal of Database Theory and Application 8(1): 169–186.

NOGRA JA, ROMANA CLS, BALAKRISHNAN E. 2020. *Baybayin* character recognition using convolutional neural network. International Journal of Machine Learning and Computing 10(2): 169–186.

NOGRA JA, ROMANA CLS, MARAVILLAS E. 2019a. Learn *baybayin*: an e-learning mobile application using a convolutional neural network. Journal of Advanced Research in Dynamical and Control Systems 11(10): 1127–1135.

NOGRA JA, ROMANA CLS, MARAVILLAS E. 2019b. Lstm neural networks for *baybayin* handwriting recognition. In: 2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS).

p. 62–66.

NOGRA JA. 2020. Inception Network for *Baybayin* Handwriting Recognition. International Journal of Advanced Trends in Computer Science and Engineering 9(1.3): 203–207.

PINO R. 2021a. Automatic transliterations of *baybayin* texts using block-level OCR. Retrieved on 15 Apr 2021 from https://tinyurl.com/hrbmm8vr

PINO R. 2021b. *Baybayin* and Latin text images. Retrieved on 04 Nov 2021 from https://tinyurl.com/xjp5tmfv

PINO R, MENDOZA R, SAMBAYAN R. 2021a. A *baybayin* word recognition system. PeerJ Computer Science 7(6): e596.

PINO R, MENDOZA R, SAMBAYAN R. 2021b. Optical character recognition system for *baybayin* scripts using support vector machine. PeerJ Computer Science 7(2): e360.

PLASENCIA MJ. 1593. *Doctrina Christiana*. Catholic Catechism, Philippines.

RAJPUT GG, UMMAPURE SB. 2017. Script identification from handwritten documents using sift method. In: 2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI). p. 520–526.

RECARIO RN, MARIANO V, GALVEZ DA, LAJARA CM. 2011. An automated reader Philippine *baybayin* scripting image processing methods. In: ICCC International Digital Design Invitation Exhibition. p. 75–76.

RECIO K, MENDOZA R. 2019. Three-step approach to edge detection of texts. Philippine Journal of Science 148(1): 193–211.

RIVERO R, KATO T. 2018. Parametric models for mutual kernel matrix completion. IEICE Transactions on Information and Systems E101.D 12: 2976–2983.

RIVERO R, LEMENCE R, KATO T. 2017. Mutual kernel matrix completion. IEICE Transactions on Information and Systems E100.D 8: 1844–1851.

SEPTIANA Y, MULYANI A, KURNIADI D, HASANUDIN H. 2021. Handwritten recognition of Hiragana and Katakana characters based on template matching algorithm. In: IOP Conference Series: Materials Science and Engineering 1098(3): 032093.

SINGH PK, SARKAR R, NASIPURI M, DOERMANN D. 2015. Word-level script identification for handwritten Indic scripts. In: 2015 13th International Conference on Document Analysis and Recognition (ICDAR). p. 1106–1110.

THOMÉ A. 2012. Svm classifiers – concepts and applications to character recognition. In: Advances in Character Recognition. Ding X ed. IntechOpen.