

A Method for Scriptio Continua Management on the Transliteration to the Balinese Script

G. Indrawan¹, I W. Sutaya¹, K. U. Ariawan¹, M. S. Gitakarma¹, I G. Nurhayata¹, I K. Paramarta²

¹Electrical Engineering and Computer Science Department

²Balinese Language Education Department

Universitas Pendidikan Ganesha (Undiksha), Singaraja, Bali, Indonesia 81116

Abstract: This research proposed a method for scriptio continua management on the learning application-based transliteration to the Balinese Script. It has not been conducted yet and is important since the proposed method eases the learning of the Balinese Script as the transliteration output. This research is a technological preservation effort to the endangered Balinese local language knowledge in Indonesia through the multi-discipline collaboration between Engineering (especially on applied Computer Science) and Language discipline. The proposed method took care of two related aspects, i.e.: (1) The Balinese Script uses scriptio continua as a style of writing without spaces, so do its transliteration result to the Balinese Script; and (2) A switching management between scriptio continua and non-scriptio continua style is needed by the learning application for the ease of learning through the visual analysis of the transliteration output which is the Balinese Script. This study was conducted on the pioneering web-based transliteration learning application, BaliScript, with the Latin text as the input and the Balinese Script as the output. Through the experiment, the proposed method gave the expected transliteration result on scriptio continua management, which can switch between three styles of writing, i.e.: (1) scriptio continua; (2) unconditional non-scriptio continua; and (3) conditional non-scriptio continua.

Keywords: Balinese Script, learning application, scriptio continua, transliteration

1. Introduction

Research in the area of Balinese Script transliteration is part of transliteration research in general (Karimi, Scholer, & Turpin, 2011; Kaur & Singh, 2014). This endangered local language, which enriches local culture, needs to be preserved its knowledge existence from all sides for a greater impact. The preservation effort by the Bali Government has already been conducted through the Bali Governor Regulation (Bali Government, 1992, 2019) and the Bali Governor Circular Letter (Bali Government, 2018). Through all of those preservation regulations, the Balinese Language (including all aspects of the Balinese Script) was considered and running as a mandatory local subject from elementary school to senior high school in Bali Province.

This research joined the preservation effort through the technological approach by proposing a method for scriptio continua management on the learning application-based transliteration to the Balinese Script. It has not been conducted yet and is important since the proposed method eases the learning through the visual analysis of the Balinese Script as the transliteration output. Moreover, this study was conducted on the pioneering web-based transliteration learning application, BaliScript, that supports Balinese Language education. The proposed method can be reused in the future for the mobile-based transliteration learning application. It is part of ubiquitous learning that provides learners with content and interaction anytime and anywhere (Hwang, Tsai, & Yang, 2008) by mobile and embedded computers with wireless networks in everyday life (Ogata, Matsuka, Bishouty, & Yano, 2009).

This research proposes a method for scriptio continua management on the learning application-based transliteration to the Balinese Script. The application uses the supporting computer font Noto Serif Balinese (NSB) with its dedicated Balinese Unicode as a standard Balinese Script font on the computer system including mobile devices (so the method can be reused on the mobile application). Moreover, this work advances the previous authors' work on the transliteration process by: (1) accommodating the rule from the Balinese Language, Script, and Literature Advisory Agency (Anom et al., 2009; Bali Government, 1992); (2) accommodating special words (G. Indrawan, Paramarta, & Agustini, 2019; G. Indrawan, Paramarta, Agustini, & Sariyasa, 2018) by using a database to avoid hard-coding a repository of special words in the application; (3) utilizing the NSB font (Google, 2020b; The Unicode Consortium, 2020b, 2020a) which is more advanced than the Noto Sans Balinese font (Google, 2020a); and (4) enhancing the learning application, that used this method, for the delivery of the transliteration and translation knowledge at the same time for the Balinese Script, Indonesian, and English (see the next Figure 2). All of those advances are considered as the contribution of this

work.

This paper is organized into several sections, i.e.: (1) the introduction section that contains the problem background in the area of the Balinese Script in general, and the transliteration to the Balinese Script in specific; (2) the related work section that describes the related works in the area of the transliteration to the Balinese Script and its scriptio continua aspect; (3) the research method section that exposes the supporting algorithm, the implementation, and the testing of the proposed method; (4) the result and discussion section that covers the analysis of the testing result; and (5) the conclusion section that consists of some important conclusion points.

2. Related Work

Several works related to scriptio continua were conducted on the Japanese to address the problem at the NLP pipelining (Murawaki & Mori, 2016), faster the morphological analyzer (Tolmachev, Kawahara, & Kurohashi, 2018); and on the Thai writing in studies of eye movements (Kasisopa, 2018; Kasisopa, Reilly, Luksaneeyanawin, & Burnham, 2016). As the Balinese Script employs its scriptio continua writing style like its relatively close (geographically and culturally) Javanese Script (Widiarti & Pulungan, 2020), several related works on Latin-to-Balinese Script transliteration were conducted on the previous author's works (Crisnapati et al., 2019; G. Indrawan, Dantes, Aryanto, & Paramarta, 2020; G. Indrawan, Gunadi, Gitakarma, & Paramarta, 2021; G. Indrawan, Paramarta, et al., 2019; G. Indrawan, Paramarta, Ramendra, Gunadi, & Sariyasa, 2021; G. Indrawan, Puspita, Paramarta, & Sariyasa, 2018; G. Indrawan, Sariyasa, & Paramarta, 2019; G. Indrawan, Setemen, Sutaya, & Paramarta, 2020; G. Indrawan, Swastika, Sariyasa, & Paramarta, 2020; Loekito, Indrawan, Sariyasa, & Paramarta, 2020). Various non-dedicated Balinese Unicode fonts, i.e. Bali Simbar (Suatjana, 1999) and Bali Simbar Dwijendra (Suatjana, 2009), and dedicated Balinese Unicode font (The Unicode Consortium, 2020b, 2020a), i.e. Noto Sans Balinese (Google, 2020a), were used for displaying Balinese Script output on those previous research. A more developed dedicated Balinese Unicode font was found on Noto Serif Balinese font (Google, 2020b) rather than Noto Sans Balinese font. Bali Simbar font was used in (G. Indrawan, Sariyasa, et al., 2019) with a relatively good accuracy result on testing cases from The Balinese Alphabet document (Sudewa, 2003). It was also used on the developed robotic system that writing Balinese Script from Latin text input (Crisnapati et al., 2019), and on the transliteration line-breaking handling exploration (G. Indrawan, Setemen, et al., 2020). Bali Simbar Dwijendra (BSD) font, as the improvement of Bali Simbar (BS) font, was used in (G. Indrawan, Swastika, et al., 2020) with additional testing cases from the Balinese Script dictionary (Anom et al., 2009), in addition to the same testing cases on (G. Indrawan, Sariyasa, et al., 2019). It was also used on the transliteration exploration of the mathematical expression (G. Indrawan, Dantes, et al., 2020). Ten transliteration lessons were also learned by using this font on the extended testing data (G. Indrawan, Gunadi, et al., 2021) other than the initial established testing data (G. Indrawan, Paramarta, et al., 2019; G. Indrawan, Sariyasa, et al., 2019). Noto Sans Balinese font was used in (G. Indrawan, Paramarta, et al., 2019) with the same testing cases in (G. Indrawan, Sariyasa, et al., 2019) and gave a relatively good accuracy result. It was also used on the developed robotic system that writing Balinese Script from Latin text input (G. Indrawan, Puspita, et al., 2018). Extensive accuracy analysis on the developed algorithm (G. Indrawan, Paramarta, et al., 2019) was conducted in (Loekito et al., 2020) for improvement in the future. Noto Serif Balinese font was used in (G. Indrawan, Paramarta, et al., 2021) for the affixed words which are the unavoidable words that need to be transliterated.

Related to the other side of transliteration, the authors' work (Gede Indrawan, Gunadi, & Paramarta, 2020) exposed the Balinese Script-to-Latin transliteration that utilizes Ocrad (Antonio Diaz, 2003) as a GNU Optical Character Recognition (OCR) tool. It is still limited only to basic syllable (Sudewa, 2003) recognition from the Balinese Script image that was based on Bali Simbar glyph shape. For advancing functionality and mobile adoption for ubiquitous learning, the authors' work (G. Indrawan, Ariawan, Agustini, & Paramarta, 2020) used the Tesseract¹ OCR tool where its data learning should be enhanced in the future.

3. Research Method

The proposed method for scriptio continua management on the learning application-based transliteration to the Balinese Script relied on two related aspects, i.e.: (1) The Balinese Script uses scriptio continua as a style of writing without spaces, so do its transliteration result to the Balinese Script; and (2) A switching management between scriptio continua and non-scriptio continua style is needed by the learning application for the ease of learning through the visual analysis of the transliteration output which is the Balinese Script.

This section describes: (1) the supporting algorithm of the proposed method; (2) the implementation on the BaliScript, which is the web-based transliteration learning application; and (3) the testing by using the updated testing cases of The Balinese Alphabet document (Sudewa, 2003) to comply with the rule from the Balinese Language, Script, and Literature Advisory Agency (Anom et al., 2009).

¹ <https://github.com/tesseract-ocr/> (Retrieved April 25, 2021)

3.1. The Algorithm

The proposed method involves the Noto Serif Balinese (NSB) font with its dedicated Balinese Unicode table (G. Indrawan, Paramarta, et al., 2021). The algorithm for scriptio continua management on the transliteration to the Balinese Script was based on two related aspects, as described previously. For the second aspect (as the core of this work), the management involves switching between three writing styles of the Balinese Script as the transliteration output, i.e.: (1) scriptio continua; (2) unconditional non-scriptio continua (with spaces); and (3) conditional non-scriptio continua without space after the consonant sign or sound killer *Adeg-Adeg* (U+1B44) “ꦲꦠꦠ꧀” (The Unicode Consortium, 2020a).

That switching between writing styles, as part of scriptio continua management, should be handled by the proposed method. Algorithm 1 shows the pseudocode and uses regular expression (Gröner & Mannicks, 2015; Hollos & Hollos, 2013) on the implementation. Noted that the Balinese code means the intermediate transformed text from a certain section of the input text (i.e. word, punctuation, etc.) during the transliteration process.

For example, the Indonesian phrase input “Bank Pembangunan Daerah Bali” (Regional Development Bank Bali) (Sudewa, 2003), which consists of sections of interest (the bold sections) related to scriptio continua management, must be transliterated to the Balinese Script as “ꦠꦤ꧀ꦧꦏ꧀ꦥꦤ꧀ꦧꦸꦁꦠꦤ꧀ꦢꦤ꧀ꦧꦶ” (scriptio continua, no space at all), “ꦠꦤ꧀ ꦧꦏ꧀ꦥꦤ꧀ꦧꦸꦁꦠꦤ꧀ꦢꦤ꧀ꦧꦶ” (unconditional non-scriptio continua), or “ꦠꦤ꧀ ꦧꦏ꧀ꦥꦤ꧀ꦧꦸꦁꦠꦤ꧀ꦢꦤ꧀ꦧꦶ” (conditional non-scriptio continua). All related aspects that accompany those three writing styles are discussed in the next section of result and discussion.

Algorithm 1. Scriptio continua management on the transliteration to the Balinese Script

```

1 if scriptio continua then
2   removes all of the spaces from the Balinese codes
3 else if unconditional non-scriptio continua then
4   preserves unconditionally all of the spaces from the Balinese codes
5 else if conditional non-scriptio continua then
6   preserves conditionally all of the spaces from the Balinese codes
   except for the spaces that come after the sound killer Adeg-Adeg

```

3.2. The Implementation

The web-based transliteration learning application, BaliScript, was constructed by the web server of Apache 2.4.46 (Win64), the database server MySQL 10.4.18-MariaDB, and programming language PHP 7.4.16 combined with JavaScript. Figure 1 shows the integration, while Figure 2 shows the example of the input and output view of the BaliScript application.

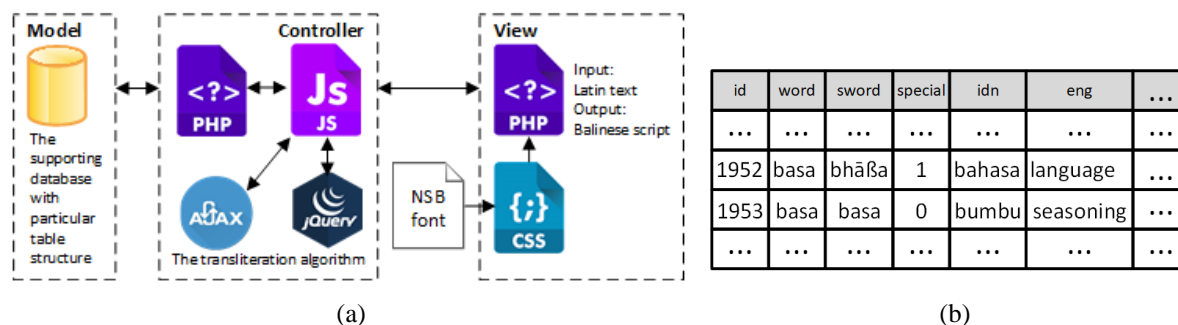
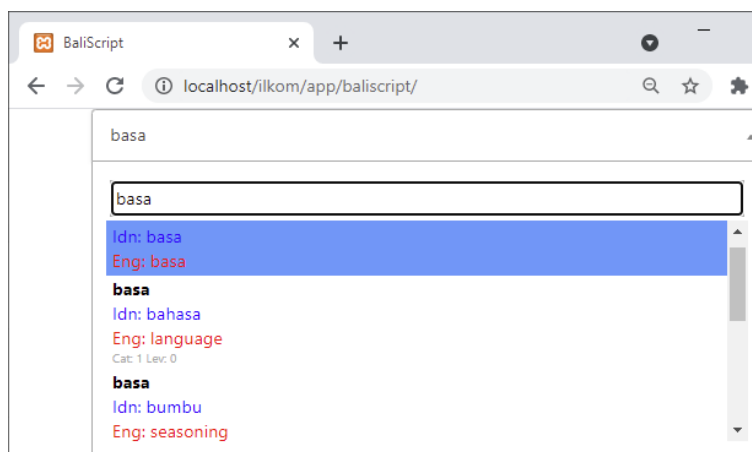


Figure 1. The implementation of the BaliScript transliteration learning application: a) MVC integration; and b) supporting database’s table

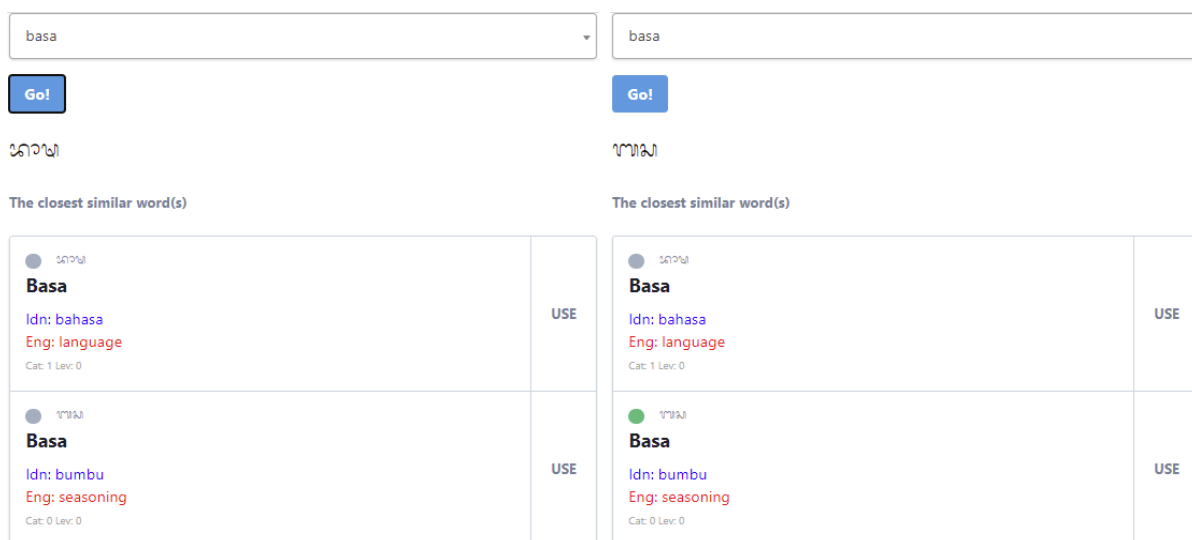
Figure 1(a) shows the integration of the Model, View, and Controller (MVC) module (Moutaouakkil & Mbarki, 2019; Singh, 2020; Sunardi & Suhajito, 2019) for the implementation of the BaliScript application. Figure 1(b) shows the supporting database’s table that consists of 35,367 records so far (Anom et al., 2009). The column “word” contains the registered word for the transliteration. If the registered word is a special word then

its related column “sword” provides the pre-defined special word (see the Balinese code in the previous section of the algorithm) that would be used for the transliteration to the Balinese Script. Additional columns “idn” and “eng”, each related to the translation in Indonesian and English for the transliterated word. As a learning application, this design is maximized for the delivery of the knowledge in the area of transliteration and also translation at the same time, specifically for the Balinese Script, Indonesian, and English. This aspect is one of the advances from the previous authors’ work and is considered as one of the contributions of this research (see the previous section of the introduction).

Figure 2(a) shows the input view with the automatic similarity ranking based on the Levenshtein distance (Aung & Htwe, 2019; Zhao & Sahni, 2019) for the entry text, whether it has already existed or not in the database. Figure 2(b) shows the output view that displays the transliteration result and other results from the closest similar words. Figure 2(c) shows the output from the other homonym word result (Roll, Correia, & Berger-Tal, 2018; Sánchez-González, 2020) by using AJAX-based switching (green indicator).



(a)



(b)

(c)

Figure 2. The BaliScript transliteration learning application with translation feature for the Balinese Script, Indonesian, and English: a) Its Select box input; b) Its output with the closest similar words; and c) Its output from the other homonym word result by AJAX switching (green indicator)

Figure 3 shows the administration panel view that accommodates special words (G. Indrawan, Paramarta, et al., 2019, 2018) by using a database’s table to avoid a hard-coded repository of special words in this learning application. This aspect is one of the advances from the previous authors’ work and is considered as one of the contributions of this research (see the previous section of the introduction).

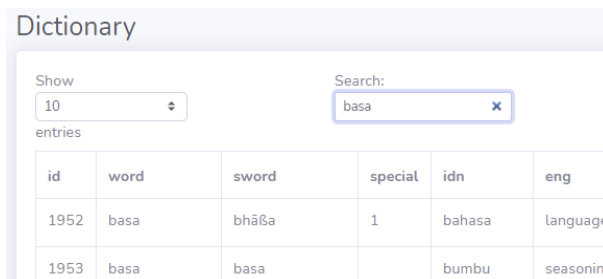


Figure 3. The administration panel view of the BaliScript transliteration learning application

3.3. The Testing

Table 1 shows the testing transliteration cases from the previous authors’ work testing data set (G. Indrawan, Paramarta, et al., 2019). Those testing cases consist of sections of interest (the bold sections) related to scriptio continua management. The analysis would be presented in the next section of result and discussion.

Noted that the testing used the updated testing cases that comply with the rule from the Balinese Language, Script, and Literature Advisory Agency (Anom et al., 2009) rather than the original testing cases from the previous authors’ work testing data set (G. Indrawan, Paramarta, et al., 2019) that refer to the Balinese Alphabet document (Sudewa, 2003). For example, since the vowel “e” of the suffix “ne” of the Balinese word “akṣarane” (that alphabets) has sound [e] (Esling, 1999), to comply with the rule, the writing of that suffix “ne” should be changed to “nē”. This aspect is one of the advances from the previous authors’ work and is considered as one of the contributions of this research (see the previous section of introduction).

Table 1. Testing transliteration cases

Case ¹	Case ²	Remarks
1 Bank Pembangunan Daerah Bali Be Pe De Bali Ba Pe Da Bali Ba Pa Da Bali	Bank Pembangunan Daerah Bali Bē, Pē, Dē, Bali Ba, Pe, Da, Bali Ba, Pa, Da, Bali	Abbreviations section in (Sudewa, 2003). Regional Development Bank Bali Be Pe De Bali Ba Pe Da Bali Ba Pa Da Bali
2 Akeh akṣarane, 47, luir ipun: akṣara suara, 14, akṣara wianjana, 33, akṣara suara punika talèr dados pangangge suara, tur madrēwe suara kakalih, kawāṣṭanin: suara hrēswa miwah dīrgha.	Akeh akṣaranē, 47, luir ipun: akṣara suara, 14, akṣara wianjana, 33, akṣara suara punika talèr dados pangangge suara, tur madrēwe suara kakalih, kawāṣṭanin: suara hrēswa miwah dīrgha.	Line Break section in (Sudewa, 2003). Many of those letters, 47, as follows vowels, 14, consonants, 33, those vowels also become vowel signs, and have two sounds, that was called: sound hrēswa and dīrgha.

¹The original testing cases from the previous authors’ work testing data set (G. Indrawan, Paramarta, et al., 2019)

²The updated testing cases that comply with the rule from the Balinese Language, Script, and Literature Advisory Agency (Anom et al., 2009)

The testing of the proposed method was conducted on the BaliScript transliteration learning application, which was run on Intel(R) Core(TM) i7-4600U CPU @ 2.09GHz platform with 8 GB RAM and Windows 8 64-bit Operating System.

4. Result and Discussion

Figure 4 shows the Balinese Script transliteration result with non-scriptio continua style, including preserved line breaks, from the BaliScript learning application. This style was supported by the Cascading Style Sheets (CSS) property white-space² that was set as *pre-line*.

This kind of non-scriptio continua style has the same space and line break format as its Latin text input from the testing transliteration cases of Table 1. Although the Balinese Script does not employ this writing style since it uses scriptio continua style, see the next Figure 5(d), this kind of non-scriptio continua style is the most useful format provided by the application for learning. It is because of its clearest mapping between the Latin text input

² https://www.w3schools.com/cssref/pr_text_white-space.asp (Retrieved April 25, 2021)

its implementation is feasible. Also, it still has a clear mapping between the Latin text input section (alphabet, syllable, word, or punctuation) and its related Balinese Script output section. That clear mapping was caused by the spaces (minus the removed spaces and line breaks) between those sections of the Latin text input that were preserved during the transliteration process. This kind of non-scriptio continua style was called conditional non-scriptio continua since the algorithm (see Algorithm 1) preserves conditionally the existing spaces.

Related to Figure 5(d), this kind of scriptio continua style has removed the space of its Latin text input from the testing transliteration cases of Table 1. It is a useful format provided by the application for learning because its implementation is feasible to show how the Balinese Script should be displayed or written. It has not a quite clear mapping between the Latin text input section (alphabet, syllable, word, or punctuation) and its related Balinese Script output section. That mapping condition was caused by the spaces between those sections of the Latin text input that were removed during the transliteration process by the algorithm (see Algorithm 1). This kind of scriptio continua style mimics Balinese Script displaying or writing in real daily life. There is a limitation on this application-based approach, i.e. the ability to fill in the empty writing space with the Balinese Script, as shown by the underlines in Figure 5(d). This limitation occurred because there is no way to distribute the Balinese Script in the writing space since no space available as a marker or separator. So, the application behaves similarly to the default left alignment on displaying or writing, and cannot be made full justify alignment (the same case if using a word processor application, as shown by Figure 6). This limitation rather defeats the essential purpose of scriptio continua style, i.e. to maximize the usage of the writing space. However, the delivery of the knowledge to introduce this kind of scriptio continua style on the application-based transliteration to the Balinese Script is considered accomplished for the learning purpose at a certain level.

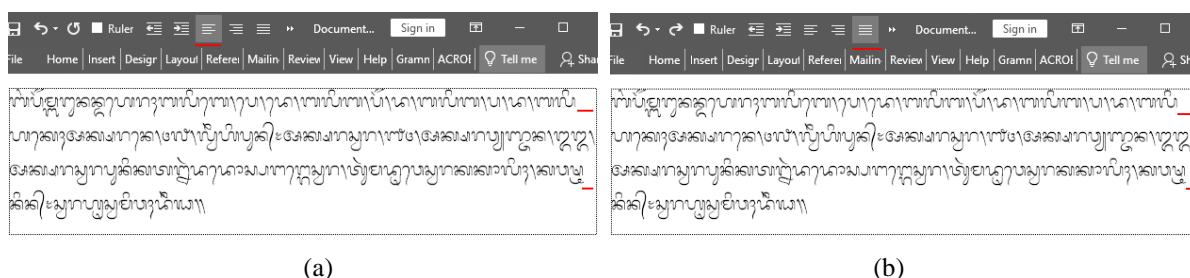


Figure 6. Displaying or writing the Balinese Script on the word processor application with: (a) left alignment; and (b) justify alignment

Conclusion

A method for scriptio continua management was proposed on the learning application-based transliteration to the Balinese Script. The management involves switching between three writing styles of the transliteration result, i.e.: (1) scriptio continua; (2) unconditional non-scriptio continua (with spaces); and (3) conditional non-scriptio continua (without space after the consonant sign *Adeg-Adeg*).

In this research, the non-scriptio continua style is not intended to replace the usage of scriptio continua style as the heart of displaying or writing the Balinese Script. It was used for the ease of learning through the visual analysis on the Balinese Script as the transliteration output from the learning application. This research added a certain perspective and strengthened the transliteration knowledge implementation through technology, which is part of the Balinese local language preservation effort.

Acknowledgment

This work was supported by the Indonesian Ministry of Education, Culture, Research, and Technology through the multi-years Applied Research Grant.

References

1. Anom, I. G. K., Suasta, I. B. M., Suardiana, I. W., Djapa, I. W., Suteja, I. W., Riken, I. M., & Swatjana, I. M. (2009). *Balinese - Indonesian Dictionary with its Latin and Balinese Script [Kamus Bali - Indonesia Beraksara Latin dan Bali]*. Denpasar: Bali Province.
2. Antonio Diaz. (2003). Ocrad - The GNU OCR. Retrieved April 25, 2021, from <https://www.gnu.org/software/ocrad/>
3. Aung, K. M. M., & Htwe, A. N. (2019). Comparison of Levenshtein Distance Algorithm and Needleman-Wunsch Distance Algorithm for String Matching. *National Journal of Parallel and Soft Computing*.
4. Bali Government. (1992). Bali Government Regulation No. 3 on Balinese Language, Script, and

- Literature [Peraturan Pemerintah Bali No. 3 tentang Bahasa, Aksara, dan Sastra Bali]. Retrieved April 25, 2021, from https://bphn.go.id/data/documents/perda_3_1992.pdf
5. Bali Government. (2018). Bali Governor Regulation No. 80 on Protection and Usage of Balinese Language, Script, and Literature, also Organizing Balinese Language Month [Peraturan Gubernur Bali No. 80 tentang Pelindungan dan Penggunaan Bahasa, Aksara, dan Sastra Bali serta Penyelenggaraan Bulan Bahasa Bali]. Retrieved April 25, 2021, from <https://jdih.baliprov.go.id/produk-hukum/peraturan/abstrak/24665>
 6. Bali Government. (2019). Bali Governor Circular Letter No. 3172 Year 2019 about The Usage of Balinese Traditional Clothing and Balinese Script [Surat Edaran Gubernur Bali No. 3172 Tahun 2019 tentang Penggunaan Busana Adat Bali dan Aksara Bali]. Retrieved April 25, 2021, from <https://jdih.baliprov.go.id/produk-hukum/peraturan/abstrak/24741>
 7. Crisnapati, P. N., Indrawan, G., Novayanti, P. D., Suranata, I. W. A., Pande, I. M. S. A., Yusadara, I. G. P. M., & Santrisna, I. P. (2019). Pasang Aksara Bot: A Balinese Script Writing Robot using Finite State Automata Transliteration Method. *Journal of Physics: Conference Series*, 1175(1), 012108. <https://doi.org/10.1088/1742-6596/1175/1/012108>
 8. Esling, J. (1999). *Handbook of the International Phonetic Association: A Guide to the Use of the International Phonetic Alphabet*. Cambridge University Press.
 9. Google. (2020a). Google Noto Fonts. Retrieved April 25, 2021, from <https://www.google.com/get/noto/#sans-bali>
 10. Google. (2020b). Google Noto Serif Balinese. Retrieved April 25, 2021, from <https://github.com/googlefonts/noto-fonts/blob/master/unhinted/ttf/NotoSerifBalinese/NotoSerifBalinese-Regular.ttf>
 11. Groner, L., & Manricks, G. (2015). *JavaScript Regular Expressions*. Birmingham: Packt Publishing.
 12. Hollos, S., & Hollos, J. R. (2013). *Finite Automata and Regular Expressions: Problems and Solutions*. Longmont, CO: Exstrom Laboratories LLC.
 13. Hwang, G. J., Tsai, C. C., & Yang, S. J. H. (2008). Criteria, strategies and research issues of context-aware ubiquitous learning. *Journal of Educational Technology and Society*, 11(2). <https://doi.org/10.1186/s40561-014-0004-5>
 14. Indrawan, G., Ariawan, K. U., Agustini, K., & Paramarta, I. K. (2020). Finite-State Machine for Post-Processing Method of Balinese Script to Latin Transliteration. *The 6th International Conference on Science, Technology, and Interdisciplinary Research*. In press
 15. Indrawan, G., Dantes, G. R., Aryanto, K. Y. E., & Paramarta, I. K. (2020). Handling of Mathematical Expression on Latin-to-Balinese Script Transliteration Method on Mobile Computing. *2020 Fifth International Conference on Informatics and Computing (ICIC)*, 1–5. IEEE. <https://doi.org/10.1109/ICIC50835.2020.9288563>
 16. Indrawan, G., Gunadi, I. G. A., Gitakarma, M. S., & Paramarta, I. K. (2021). Latin to Balinese Script Transliteration: Lessons Learned from Computer-based Implementation. *The 4th International Conference on Software Engineering and Information Management (ICSIM)*. Yokohama. In press
 17. Indrawan, G., Paramarta, I. K., & Agustini, K. (2019). A New Method of Latin-to-Balinese Script Transliteration based on Noto Sans Balinese Font and Dictionary Data Structure. *The 2nd International Conference on Software Engineering and Information Management (ICSIM)*. Singaraja. <https://doi.org/10.1145/3305160.3305167>
 18. Indrawan, G., Paramarta, I. K., Agustini, K., & Sariyasa. (2018). Latin-to-Balinese Script Transliteration Method on Mobile Application: A Comparison. *Indonesian Journal of Electrical Engineering and Computer Science*, 10(3), 1331. <https://doi.org/10.11591/ijeecs.v10.i3.pp1331-1342>
 19. Indrawan, G., Paramarta, I. K., Ramendra, D. P., Gunadi, I. G. A., & Sariyasa. (2021). A Method for the Affixed Word Transliteration to the Balinese Script on the Learning Web Application. *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*.
 20. Indrawan, G., Puspita, N. N. H., Paramarta, I. K., & Sariyasa. (2018). LBtrans-Bot: A Latin-to-Balinese Script Transliteration Robotic System based on Noto Sans Balinese Font. *Indonesian Journal of Electrical Engineering and Computer Science*, 12(3), 1247. <https://doi.org/10.11591/ijeecs.v12.i3.pp1247-1256>
 21. Indrawan, G., Sariyasa, & Paramarta, I. K. (2019). A New Method of Latin-To-Balinese Script Transliteration based on Bali Simbar Font. *2019 Fourth International Conference on Informatics and Computing (ICIC)*, 1–6. Semarang: IEEE. <https://doi.org/10.1109/ICIC47613.2019.8985675>
 22. Indrawan, G., Setemen, K., Sutaya, W., & Paramarta, I. K. (2020). Handling of Line Breaking on Latin-to-Balinese Script Transliteration Web Application as Part of Balinese Language Ubiquitous Learning. *The 6th International Conference on Science in Information Technology (ICSITech)*. Palu. <https://doi.org/10.1109/ICSITech49800.2020.9392035>
 23. Indrawan, G., Swastika, I. P. E., Sariyasa, & Paramarta, I. K. (2020). An Improved Algorithm and

- Accuracy Analysis Testing Cases of Latin-to-Balinese Script Transliteration Method based on Bali Simbar Dwijendra Font. *Test Engineering and Management*, 83, 7676–7683. Retrieved from <http://www.testmagzine.biz/index.php/testmagzine/article/view/5056>
24. Indrawan, Gede, Gunadi, I. G. A., & Paramarta, I. K. (2020). Towards Ubiquitous Learning of Balinese-to-Latin Script Transliteration as Part of Balinese Language Education. *Proceedings of the 4th Asian Education Symposium (AES 2019)*. Paris, France: Atlantis Press. <https://doi.org/10.2991/assehr.k.200513.025>
 25. Karimi, S., Scholer, F., & Turpin, A. (2011). Machine transliteration survey. *ACM Computing Surveys*, 43(3), 1–46. <https://doi.org/10.1145/1922649.1922654>
 26. Kasisopa, B. (2018). Reading without spaces between words: Eye movements in reading Thai. Dissertation
 27. Kasisopa, B., Reilly, R. G., Luksaneeyanawin, S., & Burnham, D. (2016). Child readers' eye movements in reading Thai. *Vision Research*. <https://doi.org/10.1016/j.visres.2015.07.009>
 28. Kaur, K., & Singh, P. (2014). Review of Machine Transliteration Techniques. *International Journal of Computer Applications*, 107(20), 13–16. <https://doi.org/10.5120/18866-0061>
 29. Loekito, L. H., Indrawan, G., Sariyasa, & Paramarta, I. K. (2020). Error Analysis of Latin-to-Balinese Script Transliteration Method Based on Noto Sans Balinese Font. *Proceedings of the 3rd International Conference on Innovative Research Across Disciplines (ICIRAD 2019)*. Paris, France: Atlantis Press. <https://doi.org/10.2991/assehr.k.200115.055>
 30. Moutaouakkil, A., & Mbarki, S. (2019). MVC frameworks modernization approach: Adding MVC concepts to KDM metamodel. *International Journal of Advanced Computer Science and Applications*.
 31. Murawaki, Y., & Mori, S. (2016). Wikification for scriptio continua. *Proceedings of the 10th International Conference on Language Resources and Evaluation, LREC 2016*.
 32. Ogata, H., Matsuka, Y., Bishouty, M. M. El, & Yano, Y. (2009). LORAMS: linking physical objects and videos for capturing and sharing learning experiences towards ubiquitous learning. *International Journal of Mobile Learning and Organisation*, 3(4), 337. <https://doi.org/10.1504/IJMLO.2009.027452>
 33. Roll, U., Correia, R. A., & Berger-Tal, O. (2018). Using machine learning to disentangle homonyms in large text corpora. *Conservation Biology*. <https://doi.org/10.1111/cobi.13044>
 34. Sánchez-González, J. R. (2020). Hemi-and homonyms in the big data era. *Diversity*. <https://doi.org/10.3390/d12120472>
 35. Singh, M. S. (2020). MVC Framework: A Modern Web Application Development Approach and Working. *International Research Journal of Engineering and Technology*.
 36. Suatjana, I. M. (1999). Bali Simbar. Retrieved April 25, 2021, from <http://www.babadbali.com/aksarabali/balisimbar.htm>
 37. Suatjana, I. M. (2009). *Bali Simbar Dwijendra*. Denpasar: Yayasan Dwijendra.
 38. Sudewa, I. B. A. (2003). The Balinese Alphabet. Retrieved April 25, 2021, from <http://www.babadbali.com/aksarabali/alphabet.htm>
 39. Sunardi, A., & Suharjo. (2019). MVC architecture: A comparative study between laravel framework and slim framework in freelancer project monitoring system web based. *Procedia Computer Science*. <https://doi.org/10.1016/j.procs.2019.08.150>
 40. The Unicode Consortium. (2020a). Balinese Unicode Table. Retrieved April 25, 2021, from <http://unicode.org/charts/PDF/U1B00.pdf>
 41. The Unicode Consortium. (2020b). *The Unicode Standard Version 13.0 – Core Specification* (The Unicode Consortium, Ed.). The Unicode Consortium. Retrieved from <http://www.unicode.org/versions/Unicode13.0.0/>
 42. Tolmachev, A., Kawahara, D., & Kurohashi, S. (2018). Juman++: A morphological analysis toolkit for scriptio continua. *EMNLP 2018 - Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Proceedings*. <https://doi.org/10.18653/v1/d18-2010>
 43. Widiarti, A. R., & Pulungan, R. (2020). A method for solving scriptio continua in Javanese manuscript transliteration. *Heliyon*, 6(4), e03827. <https://doi.org/10.1016/j.heliyon.2020.e03827>
 44. Zhao, C., & Sahni, S. (2019). String correction using the Damerau-Levenshtein distance. *BMC Bioinformatics*. <https://doi.org/10.1186/s12859-019-2819-0>