LBtrans-Bot: A Latin-to-Balinese Script Transliteration Robotic System based on Noto Sans Balinese Font

Article Info

Article history:

Received Jun 12th, 201x
Revised Aug 20th, 201x
Accepted Aug 26th, 201x

Keyword:

Balinese script
Latin script
Noto Sans Balinese
Robotic
Unicode

ABSTRACT

Balinese script writing, as one of Balinese cultural richness, is going to extinct because of its decreasing use. This research is one of the ways to preserve Balinese script writing using technological approach. Through collaboration between Computer Science and Balinese Language discipline, this research focused on the development of a Latin-to-Balinese script transliteration robotic system that was called LBtrans-Bot. LBtrans-Bot can be used as a learning system to give the transliteration knowledge as one aspect of Balinese script writing. In this research area, LBtrans-Bot was known as the first system that utilize Noto Sans Balinese font and was developed based on the identified seventeen kinds of special word. LBtrans-Bot consists of the transliterator web application, the transceiver console application, and the robotic arm with its GUI controller application. The transliterator used the Model-View-Controller architectural pattern, where each of them was implemented by using MySQL database (as the repository for the words belong to the seventeen kinds of special word), HTML, PHP, CSS, and Bootstrap (mostly for the User Interface responsive design), and JavaScript (mostly for the transliteration algorithm and as the controller between the Model and the View). *Dictionary* data structure was used in the transliterator memory as a place to hold data (words) from the Model. The transceiver used batch script and AutoIt script to receive and trasmit data from the transliterator to the GUI controller, which control the Balinese script writing of the robotic arm. The robotic arm with its GUI controller used open-source mDrawBot Arduino Robot Building platform. Through the experiment, LBtrans-Bot has been able to write the 34-pixel font size of the Noto Sans Balinese font from HTML 5 canvas that has been setup with additional 10-pixel length of the width and the height of the Balinese script writing area. Its transliterator gave the accuracy result up to 91% (138 of 151) testing cases of The Balinese Alphabet writing rules and examples document by Sudewa. This transliterator result outperformed the best result of the known existing transliterator based on Bali Simbar font, i.e. Transliterasi Aksara Bali, that only has accuracy up to 68% (103 of 151) cases of the same testing document. In the future work, LBtrans-Bot could be improved by: 1) Accommodating more complex Balinese script with trade off to the limited writing area of robotic system; 2) Enhancing its transliterator to accommodating the rules and/or examples from the testing document that recently cannot be handled or gave incorrect transliteration result; enriching the database consists of words belong to the seventeen kinds of special word; and implementing semantic relation transliteration.

Corresponding Author:

1. INTRODUCTION

The Balinese script, natively known as Aksara Bali or Hanacaraka, is an alphabet used in the Bali Island, Indonesia. It commonly used for writing the Austronesian Balinese language, Old Javanese, and the liturgical language Sanskrit [1]. The script is a descendant of the Brahmi script, and so has many similarities with the modern scripts of South and Southeast Asia. The Balinese script, along with the Javanese script, is considered the most elaborate and ornate among Brahmic scripts of Southeast Asia [2]. Though everyday use of the script has largely been supplanted by the Latin alphabet [3], the Balinese script has significant prevalence in many of the island's traditional ceremonies. The script is mainly used today for copying lontar or palm leaf manuscripts containing religious texts [2][4].

Based on Indonesia News Agency [3], the less use of the Balinese script has caused concern over the threat of the extinction. As Stern [5] said that saving the language can be done by cultural, political, and economic approach, another approach was taken, i.e. technological approach by this research. Through collaboration between Computer Science and Balinese Language discipline, this research focused on the development of a Latin-to-Balinese script transliteration robotic system that was called LBtrans-Bot. Transliteration itself is the conversion of a text from one script to another [6]. LBtrans-Bot can be used as a learning system to give this transliteration knowledge as one aspect of Balinese script writing. In this research area, LBtrans-Bot was known as the first system that utilize Noto Sans Balinese font [7] and was developed based on the identified seventeen kinds of special word. *Dictionary* data structure [8] was used to accomodate those seventeen kinds of special word, as the advance exploration from the thirteen kinds of special word [9].

Not so many references in this research area. Arimbawa et al. [10] has developed a Latin-to-Balinese script transliteration robotic system based on Bali Simbar font [11][12] and using C#-based desktop application to control it. On head-to-head to the LBtrans-Bot transliterator, there are several references and all of them utilized Bali Simbar font. Sartini et al. [13] has developed a text-to-digital-image converter method on Delphi-based desktop application. The output Balinese script was retrieved from pre-collected images that were captured previously from Bali Simbar font display at word processor. There are also two other methods on Android mobile application, each was called Belajar Aksara Bali (Learning Balinese Script) by Alit Jaya Trisna [14], and Transliterasi Aksara Bali (Balinese Script Transliteration) by Agus Made [15].

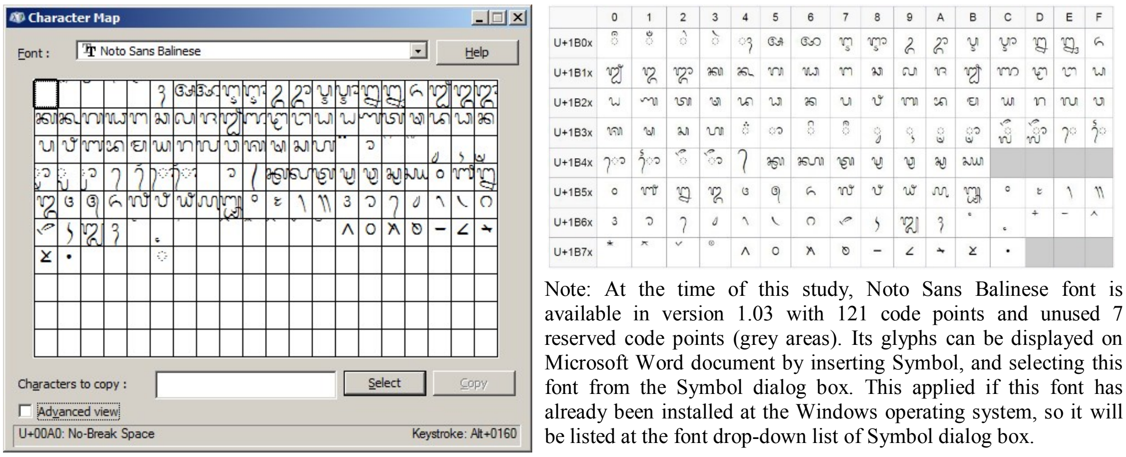
2. BALINESE SCRIPT

2.1. Noto Sans Balinese Font

Noto Sans Balinese font [7], as a smart font to accommodate Balinese Script Complex Behaviours, was released in 2014 at Google Noto homepage with hundreds of Noto fonts on the site. Noto is a font family comprising over a hundred individual fonts, which are together designed to cover all the scripts encoded in the Unicode standard. The Unicode standard [16] is the universal character encoding rules for character and written text. The Unicode standard defines a numeric value (*code point*) and a name for each character. The range of an *integer* used as a *code point* is called the *code space*. In the Unicode standard, the *code space* consists of the integers from 0 to $_{10}$FFFF^{16}, provides 1,114,112 *code points* that can be used. When referring to the *code point* in the Unicode standard, it usually uses the numeric value in hexadecimal with the prefix "U+".

As of October 2016 Noto fonts cover all 93 scripts defined in Unicode version 6.0 (released 2010), although less than 30,000 of the nearly 75,000 Chinese, Japanese and Korean (CJK) unified ideographs in version 6.0 are covered. In total Noto fonts cover nearly 64,000 characters, which is under half of the 136,755 characters defined in Unicode 10.0 (released in June 2017). Commissioned by Google, the fonts were under the Apache License 2.0 [17].

Figure 1a shows Character Map tool that was used to display Noto Sans Balinese font. By selecting certain character, its related Unicode can be seen at bottom left part of Character Map. Figure 1b shows the Unicode *code point* group allocation at U+1B00 – U+1B7F, i.e.: 1) 1B00–1B04 for various signs; 2) 1B05–1B12 for independent vowels; 3) 1B13–1B33 for consonants; 4) 1B34 for sign *rerekan*; 5) 1B35–1B43 for dependent vowel signs; 6) 1B44 for sign *adeg-adeg*; 7) 1B45–1B4B for additional consonants (Akśara Sasak); 8) 1B50–1B59 for digits; 9) 1B5A–1B60 for punctuation; 10) 1B61–1B6A for musical symbols for notes; 11) 1B6B–1B73 for diacritical marks for musical symbols; and 12) 1B74–1B7C for musical symbols. Table 1 shows several rendering process by Noto Sans Balinese font that were compared to the Bali Simbar font [11] that mostly used by the existing research in this area (see the previous section).



Note: At the time of this study, Noto Sans Balinese font is available in version 1.03 with 121 code points and unused 7 reserved code points (grey areas). Its glyphs can be displayed on Microsoft Word document by inserting Symbol, and selecting this font from the Symbol dialog box. This applied if this font has already been installed at the Windows operating system, so it will be listed at the font drop-down list of Symbol dialog box.

(a) (b)

Figure 1. Noto Sans Balinese font: a) its Character Map; b) its Unicode.

Table 1. Several rendering process differences between Noto Sans Balinese font and Bali Simbar font.

Noto Sans Balinese	Bali Simbar	Noto Sans Balinese	Bali Simbar
1		2	
b + *o* = ᬩ + ◌ᭀ = ᬩᭀ Right rendering result. Automatic positioning was applied. Involved Unicode: ᬩ ◌ᭀ U1B29 U1B40 Balinese Letter BA — Balinese Vowel Sign Taling Tedung	*b* + *o* = Wrong rendering result. *e* + *b* + *o* = ᬾ + ᬩ + ᬵ = ᬩᭀ Right rendering result by adding *e* in algorithm manually. InvolvedUnicode: ᬾ ᬩ ᬵ U0065 U0062 U006F Latin Small Letter E — Latin Small Letter B — Latin Small Letter B	*bangkung* = ᬩᬗ᭄ᬓᬸᬂ Right rendering result of word *bangkung* (pig) by automatic use of gantungan under previous consonant. Automatic identification as a consonant was also applied on consonant cluster *ng* and *ny*. Involved Unicode: ᬗ ᬓ U1B17 U1B02 Balinese Letter NGA — Balinese Letter KA ◌ᬸ ◌ᬂ U1B38 U1B02 Balinese Vowel Sign Ulu — Balinese Sign Cecek	*bangkung* = Wrong rendering result. *b\Đu** = ᬩᬗ᭄ᬓᬸᬂ Right rendering result by combination use of *b*, \, *Đ*, *u*, *and* * manually in algorithm. Automatic use of gantungan under previous consonant was not applied. Automatic identification as a consonant also was not applied on consonant cluster *ng* and *ny*. Involved Unicode: ᬗ ᬓ U005C U006B Reverse Solidus — Latin Small Letter K ◌ᬸ ◌ᬂ U0075 U002A Latin Small Letter U — Asterisk

2.2. Balinese Script Complex Behaviours

Balinese script complex behaviours demands complex rendering [18], i.e.: 1) Reordering and splitting. Some characters may have more than one separated glyph. A glyph is one pen stroke writing; 2) Various placement and shape of diacritics based on character context; 3) Contextual shaping which means glyph selection for character is determined by its neighbour character. There are various shape and position of a rendered character located above or below the other character; and 4) Complex ligature construction that was represented by new glyph as a substitution or composition of several glyphs.

Table 2 no. 1–3 shows several Balinese syllables, i.e.: 1) "ba" (U+1B29 Balinese letter *ba*); 2) "be" that comes from "ba" using *pangangge suara* "e" (U+1B3E Balinese sign *taling*). *Pangangge suara* is a Balinese vowel sign attached to a syllable. According to Simpen [19][20], *taling* is placed on the left of the syllable so that it is appeared as if it is written first and then followed by *ba*. Actually, *taling* is written later which then change the sound of “ba”. This case shows Balinese complex behaviour that requires reordering; and 3) “bo” that comes from “ba” using *pangangge suara* “o” (U+1B40 Balinese vowel sign *taling tedung*). Separated *taling* and *tedung* is written before and after the syllable, respectively. This case shows Balinese script complex behaviour that requires reordering and character splitting. *Taling tedung* is also an example of a character that has more than one separated glyph.

Table 2. Complex behaviour of the Balinese script

No	Latin	Balinese	No	Latin	Balinese	Gantungan
1	ba	ᬩ	7	děr	ᬤᭂᬃ	
2	be	ᬩᬾ	8	kra (using gantungan ra)	ᬓ᭄ᬭ	◌᭄ᬭ
3	bo	ᬩᭀ	9	skra (using gantungan ra)	ᬲ᭄ᬓ᭄ᬭ	◌᭄ᬭ
4	di	ᬤᬶ	10	krya (using gantungan rya)	ᬓ᭄ᬭ᭄ᬬ	◌᭄ᬭ᭄ᬬ
5	ding	ᬤᬶᬂ	11	na + ā = nā	ᬦ+◌ᬵ=ᬦᬵ	
6	dě	ᬤᭂ	12	ra + ya = rya	◌᭄ᬭ+◌᭄ᬬ=◌᭄ᬭ᭄ᬬ	

Table 2 no. 4–7 shows various placement and shape of diacritics based on character context, i.e.: 4) “di” that comes from “da” (U+1B24 Balinese letter *da*) using *pangangge suara* “i" (U+1B36 Balinese vowel sign *ulu*); 5) “ding” that comes from “da” using “i" and *pangangge tengenan* “ng” (U+1B02 Balinese sign *cecek*). *Pangangge tengenan* is Balinese final consonant. *Ulu* at “di” was placed in the middle above character “da”, while *ulu* at “ding” was shifted slightly by *cecek*; 6) “dě” that comes from “da” using *pangangge suara* “ě” (U+1B42 Balinese vowel sign *pepet*); and 7) “děr” that comes from “da” using “ě” and *pangangge tengenan* “r” (U+1B03 Balinese sign *surang*). *Pepet* at “dě” was placed in the middle above character “da”, while *pepet* at “děr” not only was shifted slightly by *surang* but also become smaller to make width of *pepet surang* equal to character “da” below them.

Table 2 no. 8–10 show various forms of the glyph that represent *gantungan* of Balinese syllables “ra” (U+1B2D Balinese letter *ra*). This *gantungan* is also called *cakra* or *guung*: 8) “kra” that comes from “ka” (U+1B13 Balinese letter *ka*) using *gantungan* “ra”; 9) “skra” that comes from “sa” (U+1B32 Balinese letter *sa*) and “ka” using *gantungan ra*; and 10) “krya” that comes from “ka” using *gantungan* “rya” which is combination of *gantungan* “ra” (the third *cakra*) and *gantungan* “ya” (see Table 2 no. 12). The shape of glyph *cakra* on “kra” (the first *cakra*) is narrower than the shape of glyph *cakra* on “skra” (the second *cakra*). Besides that, the glyph is written below “ka” at the end of the first *cakra*, while the glyph is written beside “ka” at the end of the second *cakra*. This case shows the Balinese script complex behaviour that some characters require glyph selection based on character context. Also, the third *cakra* shows the Balinese script complex behaviour on ligature construction.

Table 2 no. 11–12 show ligatures construction that one example was described above (Table 2 no. 12). Other case, i.e. “nā” that comes from “na” (U+1B26 Balinese letter *na*) that was followed by *pangangge suara* “ā” (U+1B35 Balinese vowel sign *tedung*).

3. RESEARCH METHOD

3.1. The architecture

A Latin-to-Balinese script transliteration robotic system based on Noto Sans Balinese font, LBtrans-Bot, consists of the transliterator web application, the transceiver console application, and the robotic arm with its Graphical User Interface (GUI) controller application, as shown by Figure 2.

The transliterator was developed based on the identified seventeen kinds of special word, valid Latin vowels, consonants, numerics, and punctuations, conversion of Latin foreign consonants, special vowels, and consecutive vowels. It used the Model-View-Controller architectural pattern. The Model was implemented by using MySQL database, as the repository for the words belong to the seventeen kinds of special word (see the next section 3.2). *Dictionary* data structure was used in the transliterator memory as a place to hold data (words) from the Model. *Dictionary* has *time complexity O*(*1*) regardless of the number of the words save inside this kind of data structure [8]. The View was implemented by using HTML, PHP, CSS, and Bootstrap, mostly for the User Interface responsive design. The Controller was implemented by using JavaScript, mostly for the transliteration algorithm and as the controller between the Model and the View.

The transceiver used batch script and AutoIt script [21] to send data from the transliterator to the GUI controller, which control the Balinese script writing of the robotic arm via USB cable or Bluetooth. The data is actually bitmap (BMP) file consists of Balinese script image that has already been thinned for the technical consideration on writing by the robotic arm. Figure 3 shows eight different patterns for the thinning process and its result on Balinese script image.

The robotic arm with its GUI controller used open-source mDrawBot Arduino Robot Building platform [22].

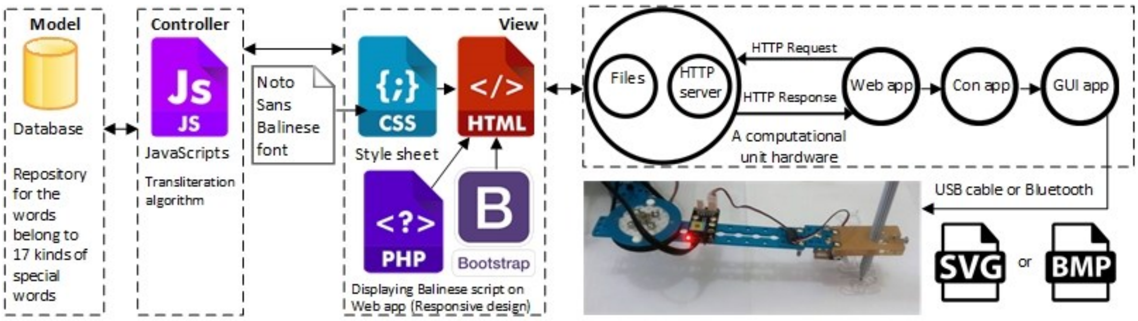
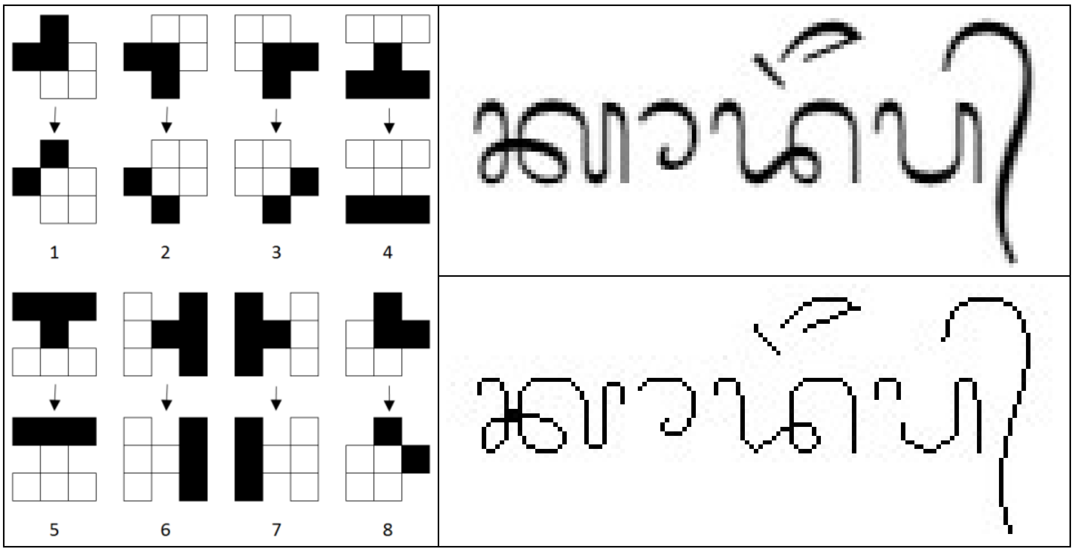


Figure 2. LBtrans-Bot architecture.



(a) (b)

Figure 3. a) Eight different patterns for the thinning process: original pattern (upper); thinned pattern (lower); b) Balinese script image: before thinning (upper); after thinning (lower).

3.2. The seventeen kinds of special word

A Latin-to-Balinese script transliteration robotic system based on Noto Sans Balinese font, LBtrans-Bot, has accommodated the seventeen kinds of special word. They were handled by LBtrans-Bot transliterator web application where the repository of those words was implemented in the Model, as shown by Figure 2. This seventeen kinds of special word were identified as part of 151 testing cases (were used for accuracy analysis of the transliterator) of The Balinese Alphabet writing rules and examples document by Sudewa [23] (see the next Table 4). This seventeen kinds of special word are listed below and more of their examples can be seen at Table 3.

1. The words where their vowel at the initial position was specifically transliterated by using the independent vowel (one of them is *akara*, U+1B05, see Figure 1b). For an example: “Akśara” (letter).

2. The words and their variants (come up due to possibility of different writing of long vowel “ā”, “ī”, “ū”, or “ö” using vowel sign *tedung*, U+1B35, see Figure 1b) that refer to one meaning and should be transliterated the same. For an example: “Kāděp” – “Kaděp” (sold).
3. The words and their vairiants (come up due to possibility of different writing of vowel “ě” using vowel sign *pepet*, U+1B42, see Figure 1b) that refer to one meaning and should be transliterated the same. For an example: “Jěro” – “Jero” (house).
4. The words and their variants (come up due to possibility of different writing between “ai” and “ê” using vowel sign *taling repa*, U+1B3F, see Figure 1b, or “au” and “ô” using vowel sign *taling repa tedung* --related to diphthong phenomen in language) that refer to one meaning and should be transliterated the same. For an example: “Daitya” – “Dêtya” (giant).
5. The words and their variants (come up due to possibility of different writing of syllable *ra repa*, i.e. “rě” or “rö”, each using U+1B0B or U+1B0C, see Figure 1b, or syllable *la lenga*, i.e. “lě” or “lö”, each using U+1B0D and U+1B0E, see Figure 1b) that refer to one meaning and should be transliterated the same. For an example: “Talěr” – “Taler” (also).
6. The words and their variants (come up due to possibility of different writing of semi-vowels “ra”, “rě”, “rö”, “ua”, “la”, and “ia”, each using U+1B44 + U+1B2D, U+1B3A, U+1B3B, U+1B44 + U+1B2F, U+1B44 + U+1B2E, and U+1B44 + U+1B2C, see Figure 1b) that refer to one meaning and should be transliterated the same. For an example: “Briag” – “Bryag” (laughter).
7. The words and their variants (come up due to possibility of different writing of *akśara şwalalita* “ṇa”, “dha”, “tha”, “ţa”, “şa”, “śa”, “gha”, “bha”, and “pha”, each using U+1B21, U+1B25, U+1B1D, U+1B23, U+1B30, U+1B31, U+1B16, U+1B2A, and U+1B28, see Figure 1b) that refer to one meaning and should be transliterated the same. For an example: “Bhiśama” – “Bhisama” (decree).
8. The words where their syllable sound must be end by using sound killer (*pangangge tengenan*) *ulu candra* (using U+1B01, see Figure 1b) or *ulu ricem* (using U+1B00, see Figure 1b), as part of *akśara modre* sign (holy symbol). For an example: “Om” (symbol of God).
9. The words and their variants (come up due to possibility of different writing of Balinese Letter Ca Laca “cha” using U+1B19, see Figure 1b) that refer to one meaning and should be transliterated the same. For an example: “Chelagi” – “Celagi” (Tamarind fruit).
10. The words where their vowel “a” at the end position can be pronounced (and written) as vowel “ě” to create their variant word. The words and their variant word refer to one meaning and should be transliterated the same. For an example: “Sěkala” – “Sěkalě” (real).
11. The words and their variants (come up due to possibility of different writing between vowel cluster “ia” and vowel-consonant cluster “iya”) that refer to one meaning and should be transliterated the same. For an example: “Kśatria” – “Kśatriya” (warrior).
12. The words with their certain single-consonant syllable and their variant word with their double-consonanst syllable, both have a single same sound for those syllables (the term for this in Balinese is *dwita*). Both words refer to one meaning and should be transliterated the same. For an example: “Utama” – “Uttama” (primary).
13. The words belong to the foreign words. For an example: “Bank”.
14. The words with assimilation combination on consonant cluster “nj” into “nyj”. For an example: “wianjana” (consonant).
15. The exception words from the rule about sound killers (*pangangge tengenan*) *cecek* (“ng” using U+1B02, see Figure 1b) or *bisah* (“h” using U+1B04, see Figure 1b) that only appears at the end of a word unless it has the same syllables, e.g. “Cengceng” (musical instrument). For an example: “Angklung” (musical instrument).
16. The words consist of *gantungan* or *gempelan* that happens very rarely when a non semi-vowel acts like a semi vowel (the term for this in Balinese is *pluta*). For an example: “Smerti” (books of Vedha).
17. The words consist of three-consonants cluster (the term for this in Balinese is *tumpuk telu*) where it is stacking *gantungan* and *gantungan* altogether in their Balinese script. Noto Sans Balinese font does not support that form, so sound killers (*pangangge tengenan*) *adeg-adeg* (using U+1B44, see Figure 1b) can be used even it is not so nice to have it in the middle of a word. For an example: “Tamblang” (a village’s name).

Table 3. More examples of the special words from The Balinese Alphabet document by Sudewa [23].

No	Special Word	Variant (*Key*)	Core (*Value*)
1	*Akśara* (letter)	akśara aksara	\akßara
	Işwara (God's name)	işwara iswara	\içwara
	Upacāra (Ceremony)	upacāra upacara	\upacāra
	Eka (One)	eka	\eka
	Airlangga (A Javanese King)	airlangga erlangga	\airlangga
	Ong (One Holy Letter)	ong	\o\m
	Om (symbol of God)	om	\au\m
	Utama (primary)	utama uttama	\utthama
2	*Kādep* (sold)	kādep kadep kādép kadép kadĕp	kādæp
	Patūt (should be)	patūt patut	patūt
	Dwī (two)	dwī dwi	dwī
	Upacāra		
	Strī (wife)	strī stri	stri\n
	Wāśţa (nama)	wāśţa wāśta wāsţa wāsta waśţa waśta wasţa wasta	wāßtta
	Dīrgha (long vowel)	dīrgha dirgha dirga	di\nrgha
3	*Kādep* (sold)		
	Jĕro (house)	jĕro jero	jæro
	Talĕr (also)	talĕr taler	talær
	Kĕrĕng (eat a lot)	kĕrĕng kereng	kæræng
	Pembangunan (Development)	pembangunan	pæmbangunan
	Drĕwe (has/have)	drĕwe drewe	dræwe
	Hrĕswa (short vowel)	hrĕswa hreswa	hræswa
4	*Daitya* (giant)	daitya detya dêtya daitia detia dêtia	daitya
5	*Talĕr* (also)		
	Kĕrĕng (eat a lot)		
5	*Drĕwe* (has/have)		
	Hrĕswa (short vowel)		
6	*Briag* (laughter)	briag briyag	byrag
7	*Akśara* (letter)		
	Işwara (God's name)		
	Gaņitri (chain)	gaņitri ganitri geņitri genitri	gaņitri
	Garudha (big eagle)	garudha garuda	garudha
	Partha (man's name)	partha	partha
	Jaţayu (a bird in Ramayana)	jaţayu jatayu	jattayu
	Bhiśama (decree)	bhiśama bhisama bisama	bhißama
	Şiwa (God's name)	şiwa siwa	çiwa
	Laghu (low tone in singing)	lagu	laghu
	Karņa (ear)	karņa karna	karnna
	Şanti (May peace be everywhere)	şanti santi	çanti
	Sukśma (thank you)	sukśma suksma	sukßma
	Kśatria (warrior)	kśatria ksatria kesatria kśatriya ksatriya kesatriya	kßatriya
	Utama (primary)		
	Dharma (religion)	dharma darma	dharmma
	Wāśţa (nama)		
	Dīrgha (long vowel)		
	Taxi (Taxi)	taxi taksi	takßi
8	*Ong* (One Holy Letter)		
	Om (symbol of God)		
	Mang (Holy Letter)	mang	ma\m
	Siddham (perfect)	siddham sidham sidam	çiddha\n
9	*Chelagi* (Tamarind fruit)	chelagi celagi	cchælagi

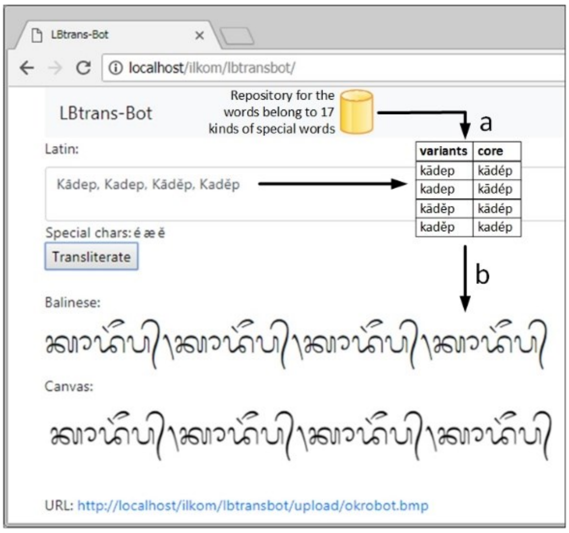
No	Special Word	Variant (*Key*)	Core (*Value*)
10	*Sekala* (*real*)	sekala sekale sekalĕ skala skale	sækala
11	*Daitya* (giant)		
	Swastiastu (May God blesses you)	swastiastu swastyastu suastiastu suastyastu	swastyastu
	Kśatria (warrior)		
	Suara (vowel)	suara swara	swara
	Quantum (Quantum)	quantum qwantum kuantum kwantum	kwantum
12	*Utama* (primary)		
	Dharma (religion)		
13	*Bank* (bank)	bank	bang
	Telefon (telephone)	telefon	telepon
	Vitamin (vitamine)	vitamin	pitamin
	Quantum (Quantum)		
	Taxi (Taxi)		
14	*Wianjana* (consonant)	wianjana wyanjana	wyanyjana
15	*Angklung* (musical instrument)	angklung	angklung
	Cengceng (musical instrument)	-	-
16	*Smerti* (books of Vedha)	smerti smrti	smrærti
17	*Tamblang* (a village's name)	tamblang	tam##blang

Note: Gray area indicates certain special words that has already listed its variant words and related core word at other previous category of special words

Severals aspects on the examples of the seventeen kinds of special word (Table 3), i.e:

1. Variant words of certain special word consist of various different wiriting of that special word. The core word of those variant words was used as the transliteration input to give the right result.
2. On the implementation, they have been put into the transliterator application memory, in the form of *dictionary* data structure, where their keys and values, each was filled with the variant and core word.
3. Certain special words can be categorized into several kinds of special word. For an example: word *Kādep* (sold) can be categorized as the second kind and the third kind of special word.

Figure 4 shows an example of the second special word that was loaded from LBtrans-Bot's repository/databse (see Figure 2) to the *dictionary* data structure. All of the various words of this example were transliterated to the same Balinese script by looking up to the core word at the *dictionary* data structure. The lookup process has *time complexity O*(*1*) regardless of the amount of words save inside this kind of data structure [8].


LBtrans-Bot
localhost/ilkom/lbtransbot/
LBtrans-Bot
Repository for the words belong to 17 kinds of special words
a
b
Transliterate

Latin:

Kādep, Kadep, Kādĕp, Kadĕp

variants	core
kādep	kādép
kadep	kādép
kādĕp	kādép
kadĕp	kadép

Special chars: é æ ĕ

Balinese:

Canvas:

URL: http://localhost/ilkom/lbtransbot/upload/okrobot.bmp

Figure 4. An example of the second kind of special word: a) at the *dictionary* data structure; b) with its transliteration result.

3.3. The algorithm

Figure 5 shows the algorithm pseudocode of a Latin-to-Balinese script transliteration robotic system based on Noto Sans Balinese font, LBtrans-Bot, which actually handles its transliterator web application and its transceiver console application (see Figure 2).

```
-- LBtrans-Bot transliterator pseudocode --
1.  Load the words of the seventeen kinds of special word from the repository into the memory (see Figure 4).
      if (a database exists)
        dictionary from MySQL database
      else
        dictionary from PHP file
2.  Lowercase the Latin string input.
3.  Filter this lowercased string by using Regular Expression (RegExp) to validate used characters.
      validVowels = “aiueoāáǎàâäæìǐīíïîǔūúùûüěēéèêëœõòǒóōö”
      validConsonants = “bcdfghjklmnpqrstvwxyzßçôņţşś”
4.  Look up the existence of this validated string at the dictionary (see Figure 4).
      if (the validated string exists)
        the looked-up string = the special string
      else if (the validated string has the longest common string with the special string)
        the looked-up string = prefix of the validated string + the special Latin string + suffix of the validated
        string
      else
        the looked-up string = the regular string
5.  Convert each of the existing foreign consonant(s) of the looked-up string to the appropriate valid consonant.
      while (“f”, “v”, “q”, “x”, or “z” exists)
        convert to “p”, “p”, “k”, “ks”, or (“j” or “s”)
6.  Convert each of the existing number(s) of the converted-foreign-consonant(s) string to the appropriate Noto
    Sans Balinese Unicode (for the next reference, it would be written as Unicode) string “&#x1B5*;” where the
    asterix sign (“*”) represents one of the number between 0 to 9 inclusively (see Figure 1b).
7.  Convert each of the existing special vowel(s) of the converted-number(s) string to to the appropriate
    Unicode string (see Figure 1b).
      while (“\n”, “\m”, “\a”, “\aa”, “\i”, “\ii”, “\u”, “\ai”, “\o”, “\au”, or “\e” exists)
        convert to the string “&#x1B00;” (ardhacandra “n”), “&#x1B01;” (candrabindu “m“), “&#x1B05;”
        (akara “a“), “&#x1B06;” (akara tedung “ā“), “&#x1B07;” (ikara “i“), “&#x1B08;” (ikara tedung
        “ī“), “&#x1B09;” (ukara “u“), “&#x1B10;” (aikara “ai“), “&#x1B11;” (okara “o“), “&#x1B12;”
        (okara tedung “au“), or “&#x1B0F;” (ekara “e“)
8.  Convert each of the existing consecutive vowel(s) of the converted-special-vowel(s) string to the
    appropriate consecutive vowel(s).
      while (“ia“,“iu“,“ii“,“ua“,“uu“,“ui“, or “ue“ exists)
        convert to “ya“,“iyu“,“iyi“,“uwa“,“uwu“,“uwi“, or “uwe“
9.  Remove each of the existing space(s) of the converted-consecutive-vowels string.
10. Add “h” at the beginning of the removed-spaces string that initialized by the vowel to become “h*”. The
    asterix sign (“*”) represents one of the vowels, i.e. “a“,“i“,“u“,“e“,“o“, or “æ“.
      while (* exists as the initial vowel)
        convert to “h*“
11. Convert each of the existing consonant(s) of the converted-initial-vowel string to the appropriate Unicode
    string of that consonant and adeg-adeg (“&#x1B**;” + “&#x1B44;”). The double asterix sign (“**”)
    represents appropriate last two digits of Unicode of the related consonant (see Figure 1b).
      while (“ng“, “ny“, “ß“, “ç“, “nn“, “dh“, “ph“, “th“, “tt“, “gh“, “kh“, “ch“, “bh“, “h“, “n“, “c“, “r“,
          “k“, “d“, “t“, “s“, “w“, “l“, “m“, “g“, “b“, “p“, “j“, or “y“ exists)
        convert to the string “&#x1B**;” + “&#x1B44;”
12. Convert each of the existing Unicode string of adeg-adeg and the following vowel (“&#x1B44;*”) of the
    converted-consonant-adeg-adeg string to the appropriate Unicode string of sign of that following vowel (see
    Figure 1b). The asterix sign (“*”) represents one of the following vowels, i.e. “ā“, “i“, “ī“, “u“, “ū“, “e“, “ai“,
    “o“, “au“, “æ“, “œ“, or “a“.
      while (“&#x1B44;*” exists)
        convert to “&#x1B35;” (tedung), “&#x1B36;” (ulu), “&#x1B37;” (ulu sari), “&#x1B38;” (suku),
        “&#x1B39;” (suku ilut), “&#x1B3E;” (taling), “&#x1B3F;” (taling repa), “&#x1B40;” (taling tedung),
        “&#x1B41;” (taling repa tedung), “&#x1B42;” (pepet), “&#x1B43;” (pepet tedung), or “”
13. Convert each of the remaining vowel(s) of the converted-adeg-adeg-vowel string to the appropriate Unicode
    string of “h” and that remaining vowel (“&#x1B33;*”). The asterix sign (“*”) represents one of the remaining
    vowel, i.e. “a“, “e“, “æ“, or “o“.
      while (* exists)
        convert to “&# x1B33;$”
```

where the dollar sign (“$”) represents the appropriate Unicode string of that remaining vowel, i.e. “”, “&#x1B3E;” (*taling*), “&#x1B42;” (*pepet*), or “&#x1B40;” (*taling tedung*).

14. Convert each of the existing Unicode string of illegal combination of “ra” + “æ”, “ra” + “œ”, “la” + “æ”, or “la” + “œ” (both of their regular and appended form) of **the converted-remaining-vowel string** to the appropriate Unicode string of *ra repa* (“ræ” or “rœ”) or *la lenga* (“læ” or “lœ”), respectively (see Figure 1b).

```
    while (* exists)
        convert to $
```

where the asterix sign (“*”) represents the Unicode string of illegal combination, i.e. “&#x1B2D;&#x1B42;” (“ra” + “æ”), “&#x1B2D;&#x1B43;” (“ra” + “œ”), “&#x1B2E;&#x1B42;” (“la” + “æ”), “&#x1B2E;&#x1B43;” (“la” + “œ”), “&#x1B44;&#x1B0B;” (*adeg-adeg* + “ræ”), “&#x1B44;&#x1B0C;” (*adeg-adeg* + “rœ”), “&#x1B44;&#x1B0D;” (*adeg-adeg* + “læ”), or “&#x1B44;&#x1B0E;” (*adeg-adeg* + “lœ”), and the dollar sign (“$”) represents the appropriate Unicode string of *ra repa* or *la lenga*, i.e. “&#x1B0B;” (“ræ”), “&#x1B0C;” (“rœ”), “&#x1B0D;” (“læ”), “&#x1B0E;” (“lœ”), “&#x1B3A;” (“ræ” sign), “&#x1B3B;” (“rœ” sign), “&#x1B3C;” (“læ” sign), or “&#x1B3D” (“lœ” sign).

Convert each of the existing Unicode string of consonants and *adeg-adeg* (“*&#x1B44;”) of **the converted-illegal-combination string** to the appropriate Unicode string of sound killer $ (see Figure 1b). The asterix sign (“*”) represents the Unicode string of consonants, i.e. “&#x1B02” (“ng“), “&#x1B03;” (“r“), or “&#x1B04;” (“h“), while the dollar sign (“$”) represents the appropriate Unicode string of sound killer, i.e. “&#x1B02;” (*cecek*), “&#x1B03;” (*surang*), or “&#x1B04;” (*bisah*).

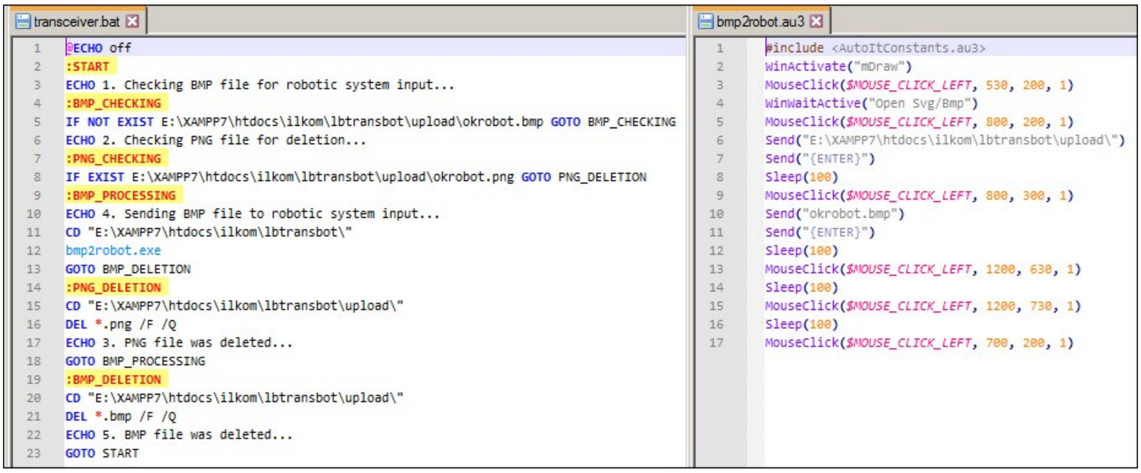
```
    while (“*&#x1B44;”)
        convert to $
```

15. Convert punctuations of **the converted-sound-killers string** to the appropriate Unicodes string, i.e. “&#x1B5A;” (*section*), “&#x1B5B;” (*honorific section*), “&#x1B5D;” (*colon*), “&#x1B5E;” (*comma*), or “&#x1B5F” (*period*).
16. Render **the converted-punctuations string** to the web application display and generate bitmap image of that string for LBtrans-Bot transceiver.

-- LBtrans-Bot transceiver pseudocode --

17. Send the bitmap image to the robotic arm by using batch script that call AutoIt script (Figure 6) to run consecutive click command on GUI controller application. This GUI controller communicate to the robotic arm via USB cable or Bluetooth (Figure 2). The whole operation illustration can be seen at the next Figure 7.

Figure 5. LBtrans-Bot algorithm pseudocode.



```
@ECHO off
:START
ECHO 1. Checking BMP file for robotic system input...
:BMP_CHECKING
IF NOT EXIST E:\XAMPP7\htdocs\ilkom\lbtransbot\upload\okrobot.bmp GOTO BMP_CHECKING
ECHO 2. Checking PNG file for deletion...
:PNG_CHECKING
IF EXIST E:\XAMPP7\htdocs\ilkom\lbtransbot\upload\okrobot.png GOTO PNG_DELETION
:BMP_PROCESSING
ECHO 4. Sending BMP file to robotic system input...
CD "E:\XAMPP7\htdocs\ilkom\lbtransbot\"
bmp2robot.exe
GOTO BMP_DELETION
:PNG_DELETION
CD "E:\XAMPP7\htdocs\ilkom\lbtransbot\upload\"
DEL *.png /F /Q
ECHO 3. PNG file was deleted...
GOTO BMP_PROCESSING
:BMP_DELETION
CD "E:\XAMPP7\htdocs\ilkom\lbtransbot\upload\"
DEL *.bmp /F /Q
ECHO 5. BMP file was deleted...
GOTO START
```

```
#include <AutoItConstants.au3>
WinActivate("mDraw")
MouseClick($MOUSE_CLICK_LEFT, 530, 200, 1)
WinWaitActive("Open Svg/Bmp")
MouseClick($MOUSE_CLICK_LEFT, 800, 200, 1)
Send("E:\XAMPP7\htdocs\ilkom\lbtransbot\upload\")
Send("{ENTER}")
Sleep(100)
MouseClick($MOUSE_CLICK_LEFT, 800, 300, 1)
Send("okrobot.bmp")
Send("{ENTER}")
Sleep(100)
MouseClick($MOUSE_CLICK_LEFT, 1200, 630, 1)
Sleep(100)
MouseClick($MOUSE_CLICK_LEFT, 1200, 730, 1)
Sleep(100)
MouseClick($MOUSE_CLICK_LEFT, 700, 200, 1)
```

Figure 6. LBtrans-Bot transceiver console application code: batch script (left); AutoIt script (right).

3.4. The testing

A Latin-to-Balinese script transliteration robotic system based on Noto Sans Balinese font, LBtrans-Bot, was tested as the integrated system consists of the transliterator web application, the transceiver console application, and the robotic arm with its Graphical User Interface (GUI) controller application (Figure 2). All of the application were run on on Intel(R) Core(TM) i5-6200U CPU @ 2.30GHz platform with 8 GB RAM and Windows 7 64-bit Operating System.

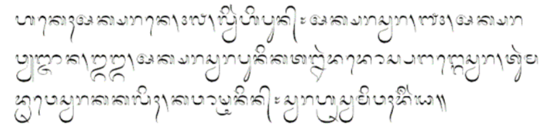
The writing on the robotic arm was tested on simple Balinese script because of the limited writing area belong to the robotic system. This succesful process reflects the succesful of the integrated system mechanism above.

At the core of LBtrans-Bot, its transliterator accuracy was tested by using The Balinese Alphabet writing rules and examples document by Sudewa [23] (Table 4a), as a project script committee related to the proposal by Eversen and Suatjana [1] for encoding the Balinese script in ISO [24]. Authors' book [25] was also used as a reference. Some of those cases referred to Simpen [19][20]. Not all of the rules can be tested independently without example (like the appended form of eighteen basic syllables at case 1–18) since provided examples are limited (case 19–25). Table 4b shows provided sentence and its transliteration for the testing case 16th (word boundaries and line break rules) at Table 4a.

Table 4. Testing cases (a); Provided sentence and its transliteration (b).

No.	Case	Remark
1st	1-18 19-25	Table 1* basic syllables (no. 1-18) Table 1* examples: word *Bakta* (bring) … *Krama* (member)
2nd	26-34	Table 2* vowel signs examples: word *Kādep* (sold) … *Dwī* (two)
3rd	35-44 45-51	Table 3* independent vowels (no. 1-10) Table 3* examples: word *Akśara* (alphabet)… *Om* (symbol of God)
4th	52-55 56-57	Table 4* illegal combination of syllable - vowel signs (no. 1-4) Table 4* examples: word *Talĕr* (therefore) … *Kĕrĕng* (eat a lot)
5th	58-62	Table 5* semi vowels examples: word *Pak Raman* (Mr. Raman) … *Briag* (laughter)
6th	63-71 72-78	Table 6* *akśara swalalita* (no. 1-9) Table 6* examples: word *Gaṇitri* (chain) … *Laghu* (low tone in singing)
7th	79-83	Table 7* sound killers examples: word *Cengceng* (musical instrument) … *Kapal* (ship)
8th	84-85	Table 8* miscellaneous signs examples: word *Mang* (holy letter) … *Siddham* (perfect)
9th	86-87	Holy symbol *Ongkara* examples: word *Om Swastiastu* (May God blesses you) … *Om Şanti, Şanti, Şanti, Om* (May peace be everywhere)
10th	88-89	Table 9* miscellaneous syllables (no. 1-2)
11th	90-99	Table 10* the digits (digit: 0 - 9)
12th	100-107	Table 11* punctuations: name *Carik* (comma) … Double Quotes
13th	 108-113 114-115 116 117-124	Some variation of usages. Combination of independence vowel *a kara* with vowel signs: vowel *i* … *ö* Pairing of *pa kapal* with *suku* or *suku ilut*: syllable *pu* … *phu* Romanization of the inherent sound: word: *Sekala* (real) Usage of *pangangge akśara*: word *Samping* (side)… *Tamblang* (a village's name)
14th	125-146	Table 13* *ligatures* (No. 1-22)
15th	147-150	Abbreviations example: word Bank *Pembangunan Daerah Bali* (Development Bank of Bali Province)… *Ba Pa Da Bali* (BPD Bali)
16th	151	Word boundaries and line break rules

(a)

No	Example
16 th	**Latin**
	Akeh akśarane, 47, luir ipun: akśara suara, 14, akśara wianjana, 33, akśara suara punika talĕr dados pangangge suara, tur madrĕwe suara kakalih, kawāśţanin: suara hrĕswa miwah dīrgha. (Many letters, 47, i.e.: vowels, 14, consonants, 33, those vowels also become vowel signs, and have two type of sounds, each was called: sound *hrĕswa* and *dīrgha*)
	Balinese
	

Note:
*refer to table number at Sudewa [23]

(b)

LBtrans-Bot, was tested as the integrated system consists of the transliterator web application, the transceiver console application, and the robotic arm with its Graphical User Interface (GUI) controller application (Figure 2). All of the application were run on on Intel(R) Core(TM) i5-6200U CPU @ 2.30GHz platform with 8 GB RAM and Windows 7 64-bit Operating System.

The writing on the robotic arm was tested on simple Balinese script because of the limited writing area belong to the robotic system. This succesful process reflects the succesful of the integrated system mechanism above.

At the core of LBtrans-Bot, its transliterator accuracy was tested by using The Balinese Alphabet writing rules and examples document by Sudewa [23], as a project script committee related to the proposal by Eversen and Suatjana [1] for encoding the Balinese script in ISO [24]. Authors' book [25] was also used as a reference. Some of those cases referred to Simpen [19][20]. Not all of the rules can be tested independently without example (like the appended form of eighteen basic syllables at case 1–18) since provided examples are limited (case 19–25).

4. RESULTS AND ANALYSIS

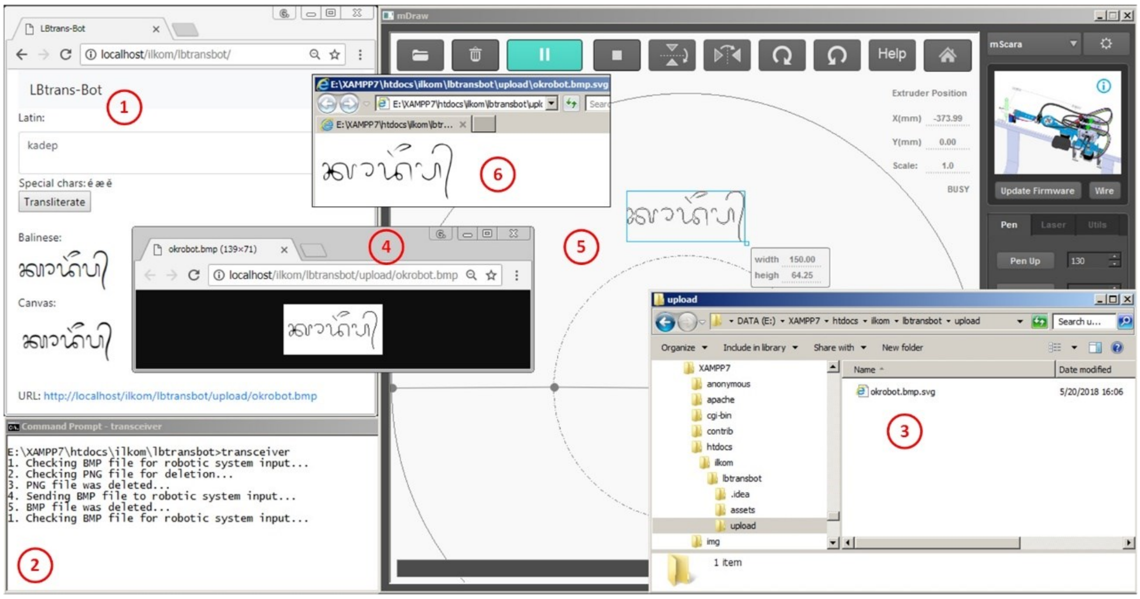
Figure 7 shows LBtrans-Bot process result with several numbered windows (see Figure 2), i.e.:

1. Windows 1 is the LBtrans-Bot transliterator web application that generates bitmap (BMP) file (see Windows 4) of an example of Balinese script (as the result of the transliteration process) and put it at certain folder (see Windows 3). The bitmap file wat setup with 34-pixel size of the Noto Sans Balinese font from HTML 5 canvas with additional 10-pixel length of the width and the height of the Balinese script writing area.
2. Windows 2 is the LBtrans-Bot transceiver console application that detects the generated bitmap file at the certain folder and executes consecutive click command on GUI controller application.
3. Windows 5 is the LBtrans-Bot Graphical User Interface (GUI) controller application communicates to the robotic arm via USB cable or Bluetooth and instructs it to write the Scalable Vector Graphics (SVG) file (see Windows 6) from the bitmap file conversion.

Table 5 shows the testing result of LBtrans-Bot transliterator, as the core of LBtrans-Bot, where column *Case* represents writing rule or example, as described by Table 4a, and column *Result* shows transliteration result whether correct or incorrect (each was represented by check and cross mark). Based on Table 5, LBtrans-Bot transliterator accuracy has passed over 91% (138 of 151) cases and outperformed Transliterasi Aksara Bali (TAB) [15]. TAB has the best accuracy result up to 68% (103 of

151) cases of the same testing document amongst the existing Latin-to-Balinese script transliteration method [9]. Figure 8 shows LBtrans-Bot transliterator result on the testing case 1st – 6th and 6th – 16th of Table 4a, where testing case numbers were displayed.

The next sixteen analysis sections, related to the sixteen testing cases (Table 4a) and several references [23][25].



URL: http://localhost/ilkom/lbtransbot/upload/okrobot.bmp

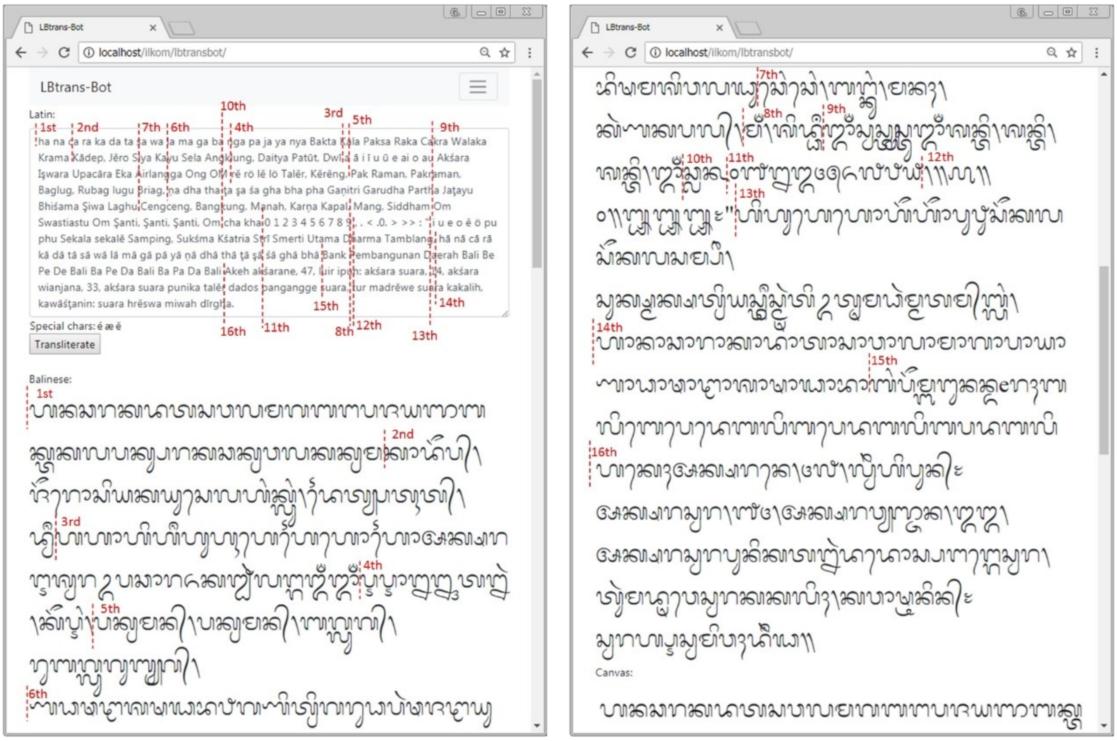
Command Prompt - transceiver

```
E:\XAMPP7\htdocs\ilkom\lbtransbot>transceiver
1. Checking BMP file for robotic system input...
2. Checking PNG file for deletion...
3. PNG file was deleted...
4. Sending BMP file to robotic system input...
5. BMP file was deleted...
1. Checking BMP file for robotic system input...
```

Figure 7. LBtrans-Bot process result.

Table 5. Testing results.

Case		Re-sult	Case		Re-sult	Case		Re-sult	Case		Re-sult	Case		Re-sult	Case		Re-sult
1th	1	√		27	√		53	√	7th	79	√		105	√		131	√
	2	√		28	√		54	√		80	√		106	√		132	√
	3	√		29	√		55	√		81	√		107	√		133	√
	4	√		30	√		56	√		82	√	13th	108	√		134	√
	5	√		31	√		57	√		83	√		109	√		135	√
	6	√		32	√	5th	58	√	8th	84	√		110	√		136	√
	7	√		33	√		59	√		85	√		111	√		137	√
	8	√		34	√		60	√	9th	86	√		112	√		138	√
	9	√	3rd	35	×		61	√		87	√		113	√		139	√
	10	√		36	×		62	√	10th	88	√		114	√		140	√
	11	√		37	×	6th	63	√		89	√		115	√		141	√
	12	√		38	×		64	√	11th	90	√		116	√		142	√
	13	√		39	×		65	√		91	√		117	√		143	√
	14	√		40	×		66	√		92	√		118	√		144	√
	15	√		41	×		67	√		93	√		119	√		145	√
	16	√		42	×		68	√		94	√		120	√		146	√
	17	√		43	×		69	√		95	√		121	√	15th	147	√
	18	√		44	×		70	√		96	√		122	√		148	×
	19	√		45	√		71	√		97	√		123	√		149	×
	20	√		46	√		72	√		98	√		124	√		150	×
	21	√		47	√		73	√		99	√	14th	125	√	16th	151	√
	22	√		48	√		74	√	12th	100	√		126	√			
	23	√		49	√		75	√		101	√		127	√			
	24	√		50	√		76	√		102	√		128	√			
	25	√		51	√		77	√		103	√		129	√			
2nd	26	√	4th	52	√		78	√		104	√		130	√			



LBtrans-Bot

Latin:

ha na ca ra ka da ta sa wa la ma ga ba nga pa ja ya nya Bakta Kala Paksa Raka Cakra Walaka Krama Kādep, Jĕro Siya Kayu Sela Angklung, Daitya Patūt, Dwīa ā i ī u ū e ai o au Akśara Işwara Upacāra Eka Airlangga Ong OM rĕ rö lĕ lö Talĕr, Kĕrĕng, Pak Raman, Pakraman, Baglug, Rubag lugu Briag, ṇa dha tha ţa şa śa gha bha pha Gaṇitri Garudha Partha Jaţayu Bhiśama Şiwa Laghu Cengceng, Bangkung, Manah, Karṇa Kapal, Mang, Siddham Om Swastiastu Om Şanti, Şanti, Şanti, Om cha kha 0 1 2 3 4 5 6 7 8 9 , . < .0. > >> : " i u e o ĕ ö pu phu Sekala sekalĕ Samping, Sukśma Kśatria Strī Smerti Utama Dharma Tamblang, hā nā cā rā kā dā tā sā wā lā mā gā pā yā ṇā dhā thā ţā şā śā ghā bhā Bank Pembangunan Daerah Bali Be Pe De Bali Ba Pe Da Bali Ba Pa Da Bali Akeh akśarane, 47, luir ipuh: akśara suara, 14, akśara wianjana, 33, akśara suara punika talĕr dados pangangge suara, tur madrĕwe suara kakalih, kawāśţanin: suara hrĕswa miwah dīrgha.

Special chars: é æ ĕ

Transliterate

Balinese:

Canvas:

Figure 8. Testing result (see Table 4a): a) the 1st – 6th case; b) the 6th –16th case.

4.1. The 1st testing case (basic syllables)

Eighteen basic syllables (*akśara wreşāstra*) and provided examples were transliterated correctly. Each syllable has an appended form (*pangangge akśara*) which kill the previous syllable sound. This appended form is called *gantungan* if hanging below the previous syllable and it is called *gempelan* if appearing after the previous syllable. Not all of this appended form can be tested independently (see The testing section). Case on Table 5 no.1 (or case 5.1) need attention since the syllable "ha" (that was written alone) by default will be transliterated correctly but its counterpart syllable "a" (that was also written alone) can be transliterated not only the same as the syllable "ha" but also can be transliterated the same as the independent vowel "a" (that was written alone) at case 5.35 (see the next section 4.3). The use is depended on the word that has the syllable "a" at the initial position. If that word is a special word **(see the 1st kind of special word at The seventeen kinds of special word section)**, for an example the word "Akśara" (alphabet) at case 5.45 (see the next section 4.3), the syllable "a" will be transliterated the same as the independent vowel "a". Otherwise, that word will be transliterated the same as the syllable "ha", for an example the word "Angklung" (musical instrument), "Akeh" (many) and "ipun" (a pronoun to the previous word "akśara") at case 5.31 (see the next section 4.2) and at Table 4b, respectively. LBtrans-Bot transliterator can handle this kind of special word through the word lookup process on *dictionary* data structure that will give average *time complexity O*(*1*) [8] regardless of the amount of words save inside this kind of data structure (see Figure 4). If certain special word is found there, LBtrans-Bot transliterator simply transliterate that word using the independent vowel. As a note, the use of the *dictionary* data structure was previously conducted by the authors for the biometric data discriminator in [26]–[29]. Unfortunately, there is still no research to know the precise list of this kind of special word that influence the accuracy of developed transliterator in general (include LBtrans-Bot transliterator). However, through the implementation of *dictionary* data structure, aggregation of known words belong to this kind of special word can be done.

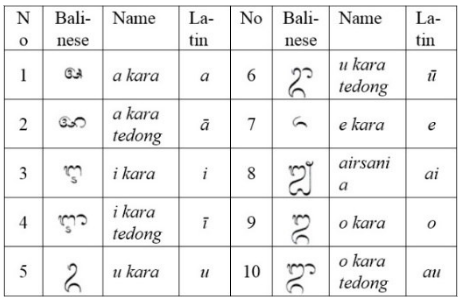
4.2. The 2nd testing case (vowel signs)

Twelve vowel signs, as part of the vowels (*akśara suara*), are attached to the syllables. Not all of these vowel signs can be tested independently, as described previously (see The testing section). All of the provided examples (case 5.26–5.34) were transliterated correctly. On case 5.26, the vowel "e" at the reference word "Kādep" (sold) should be written using the vowel "ĕ" (become "Kādêp") since *pepet* was used at reference transliteration result, like the vowel "ê" of word "Jĕro" (house) at case 5.27. Actually word “Kādĕp” and its variations (i.e. “Kādep”, “Kadĕp”, and “Kadep”) refer to one meaning and should have same transliteration. On this case, they represent another kind of special word **(see the 2nd kind of special word at The seventeen kinds of special word section)** that there is still no research to know the precise list of them. LBtrans-Bot transliterator can handle this kind of special word, basically by the same mechanism as described at the previous section 4.1. On case 5.27, actually word “Jĕro” (house) and its variation (i.e. “Jero”) refer to one meaning and should have same transliteration. They represent another kind of special word **(see the 3rd kind of special word at The seventeen kinds of special word section)** that there is still no research to know the precise list of them. LBtrans-Bot transliterator can handle this kind of special word, basically by the same mechanism as described at the previous section 4.1. On case 5.31, syllable “a” at the initial position of word “Angklung” (musical instrument) was processed by the same mechanism as syllable “a” at case 5.1 (see the previous section 4.1 that covered the analysis). On case 5.32, actually word “Daitya” (giant) and its variation (i.e. “Dêtya”) refer to one meaning and should have same transliteration. They represent another kind of special word (**see the 4th kind of special word at The seventeen kinds of special word section**) that there is still no research to know the precise list of them. LBtrans-Bot transliterator can handle this kind of special word, basically by the same mechanism as described at the previous section 4.1.

4.3. The 3rd testing case (independent vowels)

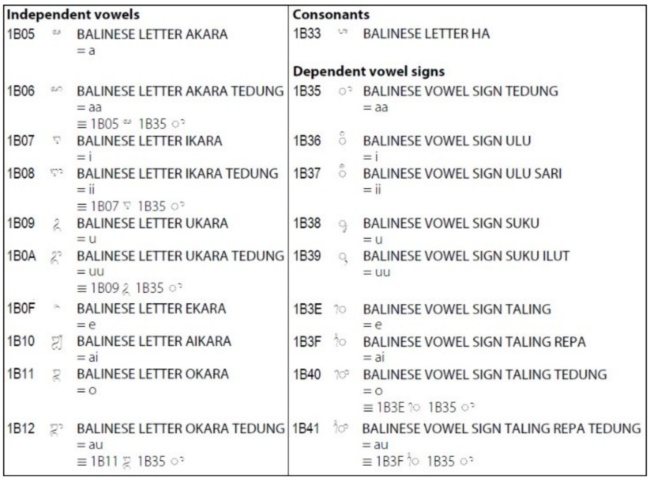
Ten independent vowels are part of the vowels (*akśara suara*), were used at the initial position of the word (case 5.35–5.51). Provided examples (case 5.45–5.51) were transliterated correctly. As described at the previous section 4.1, not all of the words with a vowel at the initial position use an independent vowel. All of the independent vowels that was written alone (case 5.35–5.44) were transliterated incorrectly (see Figure 8a) to the reference Table 6a by Sudewa [23]. Table 6b shows the Noto Sans Balinese Unicode for independently written vowels transliteration. At this time, for the sake of algoritma integrity, LBtrans-Bot transliterator prefered to translate independently written vowels by using Unicode of Consonants and Dependent vowel signs (Table 6b). Future work may address this issue.

Table 6. Independent vowels (a); Noto Sans Balinese Unicode for the vowels (b).



No	Balinese	Name	Latin	No	Balinese	Name	Latin
1	ᬅ	*a kara*	*a*	6	ᬊ	*u kara tedong*	*ū*
2	ᬆ	*a kara tedong*	*ā*	7	ᬏ	*e kara*	*e*
3	ᬇ	*i kara*	*i*	8	ᬐ	*airsania*	*ai*
4	ᬈ	*i kara tedong*	*ī*	9	ᬑ	*o kara*	*o*
5	ᬉ	*u kara*	*u*	10	ᬒ	*o kara tedong*	*au*

Note: Actually, it is optional to translate independently written vowels by using independent vowel, or syllable “ha” that was combined with dependent vowel sign (see the previous section 4.1).



Independent vowels			Consonants		
1B05	ᬅ	BALINESE LETTER AKARA = a	1B33	ᬳ	BALINESE LETTER HA
			Dependent vowel signs		
1B06	ᬆ	BALINESE LETTER AKARA TEDUNG = aa ≡ 1B05 ᬅ 1B35 ◌ᬵ	1B35	◌ᬵ	BALINESE VOWEL SIGN TEDUNG = aa
1B07	ᬇ	BALINESE LETTER IKARA = i	1B36	◌ᬶ	BALINESE VOWEL SIGN ULU = i
1B08	ᬈ	BALINESE LETTER IKARA TEDUNG = ii ≡ 1B07 ᬇ 1B35 ◌ᬵ	1B37	◌ᬷ	BALINESE VOWEL SIGN ULU SARI = ii
1B09	ᬉ	BALINESE LETTER UKARA = u	1B38	◌ᬸ	BALINESE VOWEL SIGN SUKU = u
1B0A	ᬊ	BALINESE LETTER UKARA TEDUNG = uu ≡ 1B09 ᬉ 1B35 ◌ᬵ	1B39	◌ᬹ	BALINESE VOWEL SIGN SUKU ILUT = uu
1B0F	ᬏ	BALINESE LETTER EKARA = e	1B3E	◌ᬾ	BALINESE VOWEL SIGN TALING = e
1B10	ᬐ	BALINESE LETTER AIKARA = ai	1B3F	◌ᬿ	BALINESE VOWEL SIGN TALING REPA = ai
1B11	ᬑ	BALINESE LETTER OKARA = o	1B40	◌ᭀ	BALINESE VOWEL SIGN TALING TEDUNG = o ≡ 1B3E ◌ᬾ 1B35 ◌ᬵ
1B12	ᬒ	BALINESE LETTER OKARA TEDUNG = au ≡ 1B11 ᬑ 1B35 ◌ᬵ	1B41	◌ᭁ	BALINESE VOWEL SIGN TALING REPA TEDUNG = au ≡ 1B3F ◌ᬿ 1B35 ◌ᬵ

(a) (b)

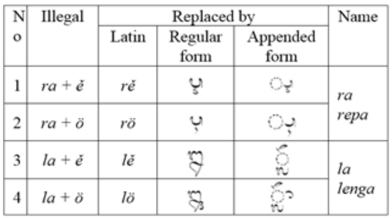
On case 5.50, the word “Ong” (a holy letter) was constructed not only by the independent vowel *o kara* “O” but also by using sound killer (*pangangge tengenan*) *ulu candra* (see the next section 4.8), instead of using default *pangangge tengenan cecek* (see the next section 4.7), to end the inherent sound of syllable “nga”. Case 5.51 is basically the same as case 5.50 but on different word “Om” (symbol of God) that was constructed not only by the independent vowel *o kara tedung* “Ô” but also by using *pangangge tengenan ulu candra* (see the next section 4.8 and 4.9), instead of using default *pangangge tengenan adeg-adeg* (see the next section 4.7) to end the inherent sound of syllable “ma”.

4.4. The 4th testing case (syllable - vowel sign combination)

Illegal combination of syllable - vowel signs happened on the case where all of the basic syllables (see the previous section 4.1) can have any of the vowel signs (see the previous section 4.2), except “ra” and “la” that each cannot have “ĕ” or “ö” appended to them. They must use regular form *ra* *repa* and *la lenga*, respectively. This illegal combination and provided examples (case 5.52–5.57) were taken care of through the correct transliteration by LBtrans-Bot transliterator (see Figure 8a). Their appended form of *ra repa* and *la lenga* cannot be tested independently, as described previously (see The testing section). As a note, there is a different glyph of "rö" at the reference Table 7a by Sudewa [23] to the Table 7b belong to the Noto Sans Balinese Unicode [7] that was used by LBtrans-Bot transliterator.

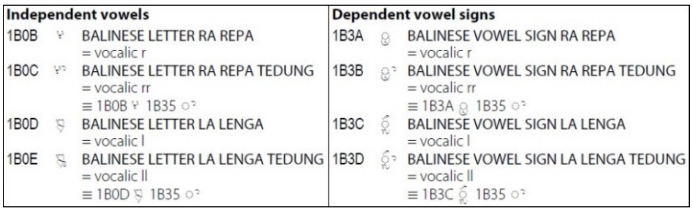
On case 5.56 that using syllable "lě", *a*ctually word "Talěr" (also) and its variation (i.e. "Taler") refer to one meaning and should have same transliteration. They represent another kind of special word (**see the 5th kind of special word at The seventeen kinds of special word section**) that there is still no research to know the precise list of them. LBtrans-Bot transliterator can handle this kind of special word, basically by the same mechanism as described at the previous section 4.1. Case 5.57 is basically the same as case 5.56 but on different syllable *"*rě". Actually word "Kěrěng" (eat a lot) and its variation (i.e. "Kereng") refer to one meaning and should have same transliteration. They represent same kind of special word like word "Talěr".

Table 7. Syllable *ra repa* and *la lenga* refer to: Sudewa document (a); Noto Sans Balinese Unicode (b).



No	Illegal	Replaced by			Name
		Latin	Regular form	Appended form	
1	*ra + ĕ*	*rĕ*	ᬋ	◌ᬺ	*ra repa*
2	*ra + ö*	*rö*	ᬌ	◌ᬻ	
3	*la + ĕ*	*lĕ*	ᬍ	◌ᬼ	*la lenga*
4	*la + ö*	*lö*	ᬎ	◌ᬽ	

(a)



Independent vowels			Dependent vowel signs		
1B0B	ᬋ	BALINESE LETTER RA REPA = vocalic r	1B3A	◌ᬺ	BALINESE VOWEL SIGN RA REPA = vocalic r
1B0C	ᬌ	BALINESE LETTER RA REPA TEDUNG = vocalic rr ≡ 1B0B ᬋ 1B35 ◌ᬵ	1B3B	◌ᬻ	BALINESE VOWEL SIGN RA REPA TEDUNG = vocalic rr ≡ 1B3A ◌ᬺ 1B35 ◌ᬵ
1B0D	ᬍ	BALINESE LETTER LA LENGA = vocalic l	1B3C	◌ᬼ	BALINESE VOWEL SIGN LA LENGA = vocalic l
1B0E	ᬎ	BALINESE LETTER LA LENGA TEDUNG = vocalic ll ≡ 1B0D ᬍ 1B35 ◌ᬵ	1B3D	◌ᬽ	BALINESE VOWEL SIGN LA LENGA TEDUNG = vocalic ll ≡ 1B3C ◌ᬼ 1B35 ◌ᬵ

(b)

4.5. The 5th testing case (semi vowels)

Four semi-vowels (*arda suara*) attached to the syllable, i.e. *guwung, suku kembung, gantungan* "la"*, and nania* for "ra", "wa" ("ua"), "la", and "ya" ("ia")*, respectively*. These appended forms cannot be tested independently, as described previously (see The testing section). Provided examples (case 5.58–5.61) were transliterated correctly.

On case 58 and 59, word "Pak Raman" (Mr. Raman) and "Pakraman" (Membership) can be transliterated the same or differently. If they were transliterated differently, syllable "ra" of word "Pak Raman" was transliterated by using basic syllable "ra" and preceded by sound killer (*pangangge tengenan*) *adeg-adeg* to form consonant "k" of word "Pakraman" (see the next section 4.7), while syllable "ra" of word "Pakraman" was transliterated by using semi-vowel *cakra*. On LBtrans-Bot transliterator, both of those words were transliterated the same by using mechanism on word "Pakraman". On case 5.62, word "Briag" should be transliterated by stacking together *cakra* and *nania* which was done by the algorithm with limitation because of flaw on Noto Sans Balinese font due to Balinese script complexity [12]. Actually word "Briag" (laughter) and its variation (i.e. "Bryag") refer to one meaning and should have same transliteration. They represent another kind of special word (**see the 6th kind of special word at The seventeen kinds of special word section**) that there is still no research to know the precise list of them. LBtrans-Bot transliterator can handle this kind of special word, basically by the same mechanism as described at the previous section 4.1.

4.6. The 6th testing case (akśara swalalita)

Nine *akśara şwalalita*, in addition to the eighteen basic syllables (see the previous section 4.1), are used for writing Kawi (Old Javanese) word. These *akśara* and provided examples (case 5.62–5.77) were transliterated correctly. Their appended form of these syllables cannot be tested independently, as described previously (see The testing section). Words using *akśara şwalalita* ("ṇa", "dha", "tha", "ţa", "şa", "śa", "gha", "bha", or "pha") refer to one meaning and should be transliterated the same. For an example on case 5.76, word "Bhiśama" (decree) and its variation (i.e. "Bhisama"). They represent another kind of special word (**see the 7th kind of special word at The seventeen kinds of special word section**) that there is still no research to know the precise list of them. LBtrans-Bot transliterator can handle this kind of special word, basically by the same mechanism as described at the previous section 4.1.

4.7. The 7th testing case (sound killers)

Four sound killers (*pangangge tengenan*), i.e. *cecek*, *surang*, *bisah*, and *adeg-adeg*, are used to end the sound of a syllable and represent consonant "ng", "r", "h", and others, respectively. *Adeg-adeg* is the default sound killer that appears after a syllable (other than "nga", "ra", and "ha"). Their appended form of these syllables cannot be tested independently, as described previously (see The testing section). Provided examples (case 5.79–5.83) were transliterated correctly. An important note on word “Angklung” (musical instrument) at case 5.31 (see the previous point 4.2) is that it is the exception from the rule said that *pangangge tengenan cecek* only appears at the end of a word unless it has the same syllables, e.g. word “Cengceng” (musical instrument) at case 5.79. On word “Angklung”, *cecek* was used for both “ng” in the middle and the end of word. The word “Angklung” represents another kind of special words (see **the 15th kind of special word at Seventeen kinds of special words section**) that there is still no research to know the precise list of them. LBtrans-Bot transliterator can handle this kind of special word, basically by the same mechanism as described at the previous section 4.1.

4.8. The 8th testing case (miscellaneous signs)

Two miscellaneous signs (part of *akśara modre*), i.e. *ulu candra* and *ulu ricem*, are kind of sound killers (see the previous section 4.7) that are used to write Sanskrit words, usually part of prayers. They are used to end the sound of a syllable and represent consonant “ng” and “m”, respectively (at the previous section 4.7, their counterpart sound killer of “ng” and “m” is *cecek* and *adeg-adeg*, respectively). Their appended form of these syllables cannot be tested independently, as described previously (see The testing section). All of provided examples (case 5.84–5.85) were transliterated correctly. On case 5.84, word “Mang” (a holly letter) represent another kind of special word (**see the 8th kind of special word at The seventeen kinds of special word section**) that there is still no research to know the precise list of them. On case 5.85, word *Siddham* (perfect) represent same kind of special word like word *Mang* that there is still no research to know the precise list of them. LBtrans-Bot transliterator can handle this kind of special word, basically by the same mechanism as described at the previous section 4.1.

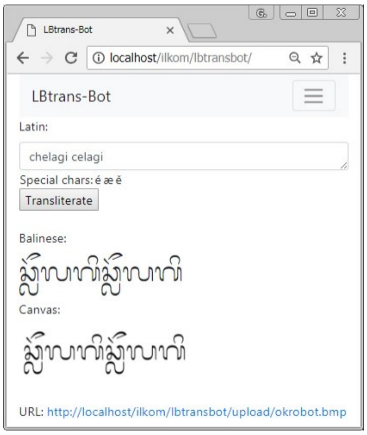
4.9. The 9th testing case (holy symbol)

When independent vowel *au kara* (see the previous section 4.3) meets sound killer *ulu candra* (see the previous section 4.8), the Romanization is not “Aung”, but “Om”. “Om” has a special name, i.e. *Ongkara*, as a holy symbol (*akśara modre*). This word is used almost everywhere in the text, as it is the symbol of God Himself. The most notable sentences using “Om” are the greetings. All of provided examples (case 5.86–5.87) were transliterated correctly, i.e. phrase “Om Swastiastu” (May God blesses you) and “Om Şanti, Şanti, Şanti, Om” (May peace be everywhere), respectively.

On case 5.86, word “Om” represent same kind of special word (**see the 8th kind of special word at The seventeen kinds of special word section**) like word “Mang” (see the previous section 4.8). Word “Swastiastu” was transliterated by stacking together cakra and *nania* (see the previous section 4.5). Actually, word “Swastiastu” and its variation (i.e. “Swastyastu”) refer to one meaning and should have same transliteration. They represent same kind of special word (**see the 6th kind of special word at The seventeen kinds of special word section**) like word “Briag” (laughter) at case 5.62 (see the previous section 4.5). On case 5.87, actually, word “Şanti” and its variation (i.e. “Santi”) refer to one meaning and should have same transliteration. They represent same kind of special word (**see the 7th kind of special word at The seventeen kinds of special word section**) like word “Şiwa” (God’s name) at case 5.77 (see the previous section 4.6).

4.10. The 10th testing case (miscellaneous syllables)

The existence of two miscellaneous syllables (that apparently borrowed from Javanese) in Balinese script is very rare (case 5.88–5.89). They were transliterated correctly. Their appended form of these syllables cannot be tested independently, as described previously (see The testing section). On case 5.88, the syllable “cha” has no regular form. It is always paired with the normal form of “ca”. Figure 9 shows an example using this syllable. Actually word “Chelagi” (Tamarind fruit) and its variation (i.e. “Celagi”) represent another kind of special word (**see the 9th kind of special word at The seventeen kinds of special word section**) that there is still no research to know the precise list of them. LBtrans-Bot transliterator can handle this kind of special word, basically by the same mechanism as described at the previous section 4.1.



LBtrans-Bot

Latin:

chelagi celagi

Special chars: é æ ĕ

Transliterate

Balinese:

Canvas:

URL: http://localhost/ilkom/lbtransbot/upload/okrobot.bmp

Figure 9. Ca Laca Transliteration.

4.11. The 11th testing case (digits)

All of the digits (0–9) were transliterated correctly (case 5.90–5.99).

4.12. The 12th testing case (punctuations)

All of independently written punctuations (case 5.100–5.107) were transliterated correctly. Comma (,), period (.), less-than (<), period-0-period (.0.), greater-than (>), double greater-than (>>), and colon (;) sign was transliterated correctly become *carik* (case 5.100), *carik pareren* (case 5.101), *panten* (case 5.102), *pasalinan* (case 5.103), *pamada* (case 5.104), *carik agung* (case 6105), and *carik pamungkah* (case 5.106) sign, respectively. Double quotes (case 5.107) has the same sign ("). *Panten* is used at the beginning of a letter, a story, or a verse, while *pasalinan* is at the end of it. *Pamada* is used at the beginning of a religious text, while *carik agung* is at the end of it.

4.13. The 13th testing case (some variation of usages)

Some variation of usages includes: 1) incorrect combination of independence vowel *a kara* (see the previous section 4.3) and vowel signs (see the previous section 4.2); 2) special use of syllable *pa kapal* (see the previous section 4.6) that is never attached to *suku* or *suku ilut* (see the previous section 4.2); 3) romanization of the inherent sound; and 4) the use of *pangangge akśara* (see the previous section 4.1).

On the first variation of usages, for any vowel sounds, there are independent vowel glyphs that are ready to be used. On case 5.108–5.111, incorrect combination of independence vowel *a kara* and vowel sign were shown by using *a kara* combined with *ulu*, *suku*, *taling,* and *taling-tedung* at case 5.108, 5.109, 5.110, and 5.111, respectively. None of those incorrect combination came up on LBtrans-Bot transliterator. On case 5.112–5.113, vowel sound "ĕ" and "ö" that don't have independent vowel form, should be written by using syllable "ha" combined with *pepet* and *pepet-tedung* at case 5.112 and 5.113, respectively. Vowel sound "ĕ" and "ö" were transliterated correctly.

On the second variation of usages, *pa kapal* can be paired with any vowel signs but its shape is not the same as other syllables since its final stroke is not going down. Hence *suku* and *suku ilut* cannot be attached to *pa kapal* but they are can positioned below it. On case 5.114–5.115, where each of syllable "pa" and *pa kapal* was paired with *suku*, both of them were transliterated correctly by hAksara.

On the third variation of usages, a stand-alone syllable has inherent sound that is always Romanized as "a" and it is common to a Balinese people to pronounced an "a" at the end of a word as "ĕ". On case 5.116, word "Sekala" and "sekalĕ" was transliterated correctly. For both of the words, vowel "e" at reference word "sekala" should be written using vowel "ĕ" (become "sĕkala") since vowel sign *pepet* was used at reference transliteration result, like vowel "ĕ" of word "Jĕro" at case 27 (see the previous section 4.2). Actually, word "Sekala" (real) and its variations (i.e. "Sekalĕ", "Sĕkala", and "Sĕkalĕ") refer to one meaning and should have same transliteration. They represent another kind of special word (**see the 10th kind of special word at The seventeen kinds of special word section**) that there is still no research to know the precise list of them. LBtrans-Bot transliterator can handle this kind of special word, basically by the same mechanism as described at the previous section 4.1.

On the fourth variation of usages, all provided examples (case 5.117–5.124) were transliterated correctly. Several notes on word "Sukśma" (thank you), "Kśatria" (warrior), "Strī" (wife), "Smerti" (books of Veda), "Utama" (primary), and "Dharma" (religion). First four examples are form of *gantungan/gempelan* happens very rarely, when a non-semi-vowel acts like a semi vowel (see the previous section 4.5). The term for this in Balinese is *pluta*. Last two examples have double consonant syllable, but actually a single sound. This occurrence is called *dwita*. On case 5.117, consonant "m" of word "Samping" (side) was transliterated correctly without using *pangangge tengenan adeg-adeg* in the middle of the word. Even though using *adeg-adeg* in the middle of the word is also correct, the preferred form is the one without *adeg-adeg* for the sake of aesthetic aspect. On case 5.118, consonant "k" of word "Sukśma" was transliterated correctly without using *adeg-adeg* in the middle of the word. On case 5.119, word "Kśatria" was transliterated correctly. This is the case where vowel cluster "ia" was not transliterated by using vowel sign (see the previous section 4.2) or by changing it first become "ya" (see the next section 4.5). Actually, word "Kśatria" and its variations (i.e. "Ksatria", "Kśatriya", and "Ksatriya") refer to one meaning and should have same transliteration. They represent another kind of special word (**see the 11th kind of special word at The seventeen kinds of special word section**) that there is still no research to know the precise list of them. LBtrans-Bot transliterator can handle this kind of special word, basically by the same mechanism as described at the previous section 4.1. On case 5.121, vowel "e" at reference word "Smerti" should be written by using vowel "ĕ" (become "Smĕrti"*)* since gantungan "mĕ" was used at reference transliteration result. This gantungan "mĕ" of word "Smĕrti" was transliterated correctly. Actually word "Smerti" and its variation (i.e. "Smrti") refer to one meaning and should have same transliteration. They represent another kind of special words (see **the 16th kind of special word at Seventeen kinds of specials words section**) that there is still no research to know the precise list of them. LBtrans-Bot transliterator can handle this kind of special word, basically by the same mechanism as described on the special word at the previous section 4.1. On case 5.122, actually word "Utama" and its variation (i.e. "Utthama") refer to one meaning and should have same transliteration. They represent another kind of special word (**see the 12th kind of special word at The seventeen kinds of special word section**) that there is still no research to know the precise list of them. LBtrans-Bot transliterator can handle this kind of special word, basically by the same mechanism as described at the previous section 4.1. On case 5.123, actually word "Dharma" and its variation (i.e. "Dharmma"*)* refer to one meaning and should have same transliteration. They represent same kind of special word like word "Utama" that there is still no research to know the precise list of them.

4.14. The 14th testing case (ligatures)

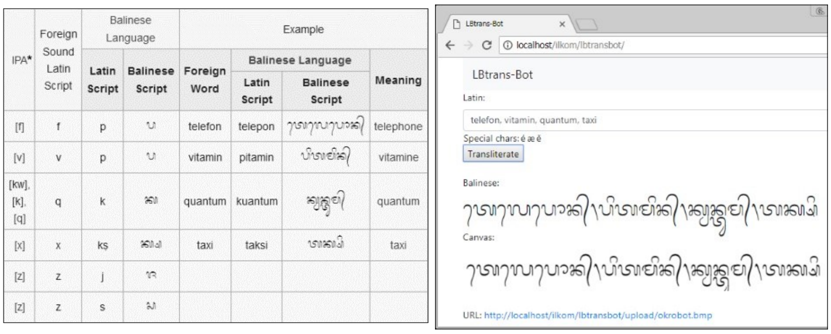
The *ligature* as one pen strokes of two glyphs is desirable but not mandatory form. *Tedung* forms *ligature* with the certain syllable. *Suku* and *suku ilut* form *ligature* with the certain *gantungan* or *gempelan* (see the previous section 4.1). On the case 5.125–5.146, *tedung ligatures* were not rendered by LBtrans-Bot transliterator, instead tedung was rendered separately from preceding syllable, which is also correct. *Ligature* with g*empelan* "pa", "sa", "ya", *sa sapa*, *suku,* or *suku ilut* cannot be tested independently, as described previously (see The testing section).

4.15. The 15th testing case (abbreviations)

Three different scheme for abbreviations can be used in Balinese, i.e.: 1) The one endorsed by the government to abbreviate government institutions. The scheme is to follow the way the abbreviation pronounced in Indonesian language; 2) The one used by I Wayan Simpen A.B. in his schoolbook [20]. The scheme is to use the first syllable with all the vowel signs attached to it; or if it is an independent vowel, then the independent vowel itself is used; and 3) The one less commonly used, but somehow the shortest one. The scheme is to use only syllable or independent vowel.

Case 5.147–5.150 consists of phrase "Bank Pembangunan Daerah Bali" (Development Bank of Bali Province) and all of its three abbreviation schemes. Latin abbreviation of that phrase is "BPD Bali" and its three abbreviation schemes, i.e. "Be Pe De Bali", "Ba Pe Da Bali", and "Ba Pa Da Bali" at case 5.148, 5.149, and 5.150, respectively. On all of abbreviations schema there are *cariks* (see the previous section 4.12) between syllable and/or word that still cannot be accommodated by LBtrans-Bot transliterator (neither do other algorithms) since there is no smart way to differentiate between this abbreviation phrase with non-abbreviation phrase, except all of these abbreviation phrases were manually put into the database (see Figure 2), which are huge in number.

On case 5.147, phrase "Bank Pembangunan Daerah Bali" was transliterated correctly. Actually, word "Bank" represents another kind of special word (**see the 13th kind of special word at The seventeen kinds of special word section**) that there is still no research to know the precise list of them. LBtrans-Bot transliterator can handle this kind of special word, basically by the same mechanism as described at the previous section 4.1. Related to these special words, LBtrans-Bot transliterator has also provided algorithm (**see the 5th step of psedocode at The algorithm section**) to handle foreign sound transliteration, as shown by Figure 10a [30]. Figure 10b shows foreign words transliteration result.



IPA*	Foreign Sound Latin Script	Balinese Language		Example			
		Latin Script	Balinese Script	Foreign Word	Balinese Language		Meaning
					Latin Script	Balinese Script	
[f]	f	p	ᬧ	telefon	telepon	ᬢᬾᬮᬾᬧᭀᬦ᭄	telephone
[v]	v	p	ᬧ	vitamin	pitamin	ᬧᬶᬢᬫᬶᬦ᭄	vitamine
[kw], [k], [q]	q	k	ᬓ	quantum	kuantum	ᬓ᭄ᬯᬦ᭄ᬢᬸᬫ᭄	quantum
[x]	x	kṣ	ᬓ᭄ᬱ	taxi	taksi	ᬢᬓ᭄ᬱᬶ	taxi
[z]	z	j	ᬚ				
[z]	z	s	ᬲ				

(a)

LBtrans-Bot

Latin:

telefon, vitamin, quantum, taxi

Special chars: é æ ě

Balinese:

ᬢᬾᬮᬾᬧᭀᬦ᭄᭞ᬧᬶᬢᬫᬶᬦ᭄᭞ᬓ᭄ᬯᬦ᭄ᬢᬸᬫ᭄᭞ᬢᬓ᭄ᬱᬶ

Canvas:

ᬢᬾᬮᬾᬧᭀᬦ᭄᭞ᬧᬶᬢᬫᬶᬦ᭄᭞ᬓ᭄ᬯᬦ᭄ᬢᬸᬫ᭄᭞ᬢᬓ᭄ᬱᬶ

URL: http://localhost/ilkom/lbtransbot/upload/okrobot.bmp

(b)

Figure 10. Transliteration result of foreign words: a) the reference; b) the result.

Still on case 5.147, word "Pembangunan" and "Daerah" is Indonesian words and since no different in writing vowel "e" for different pronunciation "ĕ" (belong to "e" at "Pembangunan") and "e" (belong to "e" at "Daerah"), it is become difficult on transliteration. LBtrans-Bot transliterator can handle this by relating them to the previous special word like word "Jĕro" (house) at case 5.27 of the previous section 4.2 (**see the 3rd kind of special word at The seventeen kinds of special word section**). Case 5.149 is the same as case 5.147 where vowel "e" at phrase "Ba Pe Da Bali" is pronounced as "ĕ".

4.16. The 16th testing case (word boundaries and line break rules)

There are no spaces to separate words in Balinese script. In the old time of writing on dried palm leaves (it was called *lontar*), spaces were scarce and the "page setup" for *lontar* was always a thin landscape. The number of lines is small, with every space must be filled for optimal use. There was common practice to break the sentence at any places. For modern writing, the following rules of thumb should apply, i.e.: 1) No line breaks allowed between syllable and any of its signs; and 2) No line breaks allowed just before a colon, comma, or full stop. On case 5.151, sentence was transliterated correctly. Several notes on several words in the sentence, i.e. "Akeh" (Many), "akśarane" (those alphabets), "luir" (i.e.), "ipun" (a pronoun to previous word akśarane), "suara" (vowel), "wianjana" (consonant), "madrĕwe" (have), "kawāśţanin" (being named), "hrĕswa" (short vowel), and "dīrgha" (long vowel).

On word "Akeh", vowel "A" was transliterated correctly without using independent vowel (see the previous section 4.1). On word "akśarane", LBtrans-Bot transliterator basically can handle that word (**see the 4th step of psedocode at The algorithm section**) which is basically constructed by basic word "akśara" (see the previous section 4.3) and suffix "ne". On word "luir", semi vowel "ua" (that construct "ui" by using vowel sign *ulu*) was transliterated correctly (see the previous section 4.5). Actually, word "luir" and its variation (i.e. "lwir") refer to one meaning and should have same transliteration. They represent same kind of the special word like word "Briag" (laughter) at case 5.62 (see the previous section 4.5). On word "ipun", vowel "i" was transliterated correctly without using independent vowel (see the previous section 4.1). On word "suara", semi vowel "ua" was transliterated correctly (see the previous section 4.5). Actually, word "suara" and its variation (i.e. "swara") refer to one meaning and should have same transliteration. They represent same kind of special word like word "Briag" (intense) at case 5.62 (see the previous section 4.5). On word "wianjana", semi vowel "ia" was transliterated correctly (see the previous section 4.5). At another aspect, cluster "nj" was transliterated correctly using gantungan "ja" on syllable "nya", instead on syllable "na". This is because there is assimilation combination on syllable "na" into syllable "nya" [31]. Actually, word "wianjana" and its variations (i.e. "wyanjana") refer to one meaning and should have same transliteration. They represent another kind of special word (**see the 14th kind of special word at The seventeen kinds of special word section**) that there is still no research to know the precise list of them. LBtrans-Bot transliterator can handle this kind of special word, basically by the same mechanism as described at the previous section 4.1. On word "madrĕwe", vowel "ĕ" was transliterated correctly. Actually word "madrĕwe" and its variation (i.e. "madrewe") refer to one meaning and should have same transliteration. They represent same kind of special word like previous word *Kādĕp* (sold) at case 5.26 (see the previous section 4.2). On word "kawāśţanin", LBtrans-Bot transliterator basically can handle that word (**see the 4th step of psedocode at the algorithm section**) which is basically constructed by basic word "wāśţa" (see the previous section 4.6), preffix "ka", and suffix "nin". Actually, word "kawāśţanin" and its variation (i.e. "kawastanin") refer to one meaning and should have same transliteration. They represent same kind of special word like word "Jaţayu" (a bird) at case 5.75 (see the previous section 4.6). On word "hrĕswa", vowel "ĕ" was transliterated correctly. Actually word "hrĕswa" and its variation (i.e. "hreswa") refer to one meaning and should have same transliteration. They represent same kind of special word like the previous word *Kādĕp* (sold) at case 5.26 (see the previous section 4.2). On word "dīrgha", vowel "ī" and "gha" (related to vowel sign *ulu sari* and syllable *ga gora*, see the next section 4.2 and 4.6, respectively) was transliterated correctly. Actually, word "dīrgha" and its variation (i.e. "dirgha") refer to one meaning and should have same transliteration. They represent same kind of special word like word "Laghu" (song) at case 5.78 (see the previous section 4.6).

On another case, period sign (.) should be written at the end of this sentence since punctuation *carik pareren* was used at the reference transliteration result (see the previous section 4.12).

5. CONCLUSION

A Latin-to-Balinese script transliteration robotic system based on Noto Sans Balinese font [7], LBtrans-Bot, has been developed and has been able to write the 34-pixel font size of the Noto Sans Balinese font from HTML 5 canvas that has been setup with additional 10-pixel length of the width and the height of the Balinese script writing area. Through the comprehensive accuracy analysis, its transliterator web application gave the accuracy result up to 91% (138 of 151) cases of The Balinese Alphabet writing rules and examples document by Sudewa [23], as a project script committee related to the proposal by Eversen and Suatjana [1] for encoding the Balinese script in ISO [24]. This transliterator result outperformed the best result of the known existing transliterator based on Bali Simbar font [11][12], i.e. Transliterasi Aksara Bali [15], that only has accuracy up to 68% (103 of 151) cases of the same testing document [9].

In the future work, LBtrans-Bot could be improved by: 1) Accommodating more complex Balinese script with trade off to the limited writing area of robotic system; 2) Enhancing its transliterator to accommodating the rules and/or examples from the testing document that recently cannot be handled or gave incorrect transliteration result (like the 3rd testing case of independent vowels where at this time, for the sake of algoritma integrity, the transliterator prefered to translate independently written vowels by using Unicode of Consonants and Dependent vowel signs, as shown by Table 6b); Enriching the database (Figure 2) consists of words belong to the seventeen kinds of special word by using identified those words from the existing Balinese script dictionary from Bali Province government [32]; and Implementing semantic relation transliteration, for an example word “Om” (**see the 8th kind of special word at The seventeen kinds of special word section**) beside means as symbol of God, also means as uncle. It depends on the semantic context.

ACKNOWLEDGEMENTS

This research was supported by Indonesian Ministry of Research, Technology, and High Education through Grant No. 113/UN48.15/LT/2018.

The transliterator web application module of this LBtrans-Bot has already been hosted at server of Computer Science Graduate Program of Author institution as a JavaScript-based web application for the purpose of learning and testing for the further improvement [33].

REFERENCES

[1] M. Eversen and I. M. Suatjana, “N2908: Proposal for Encoding the Balinese Script in the UCS.” 2005.

[2] J. Kuipers, “Indic Scripts of Insular Southeast Asia: Changing Structures and Functions.” Tokyo University of Foreign Studies, Tokyo, 2003.

[3] Antara, “Satu Hari Berbahasa Bali | The Indonesia News Agency,” 08-Dec-2009. [Online]. Available: http://bali.antaranews.com/berita/2756/satu-hari-berbahasa-bali. [Accessed: 15-Jun-2017].

[4] R. Fox, “Rival Styles of Writing, Rival Styles of Practical Reasoning.” Heidelberg Institut für Ehtnologie, Heidelberg, 2013.

[5] A. Stern, “An Community in Promoting the Balinese Language | BASAbali @ ARMA,” 2017. [Online]. Available: http://basabali.org/. [Accessed: 15-Jun-2017].

[6] N. S. Kharusi and A. Salman, “The English Transliteration of Place Names in Oman,” *Journal of Academic and Applied Studies*, vol. 1, no. 3, pp. 1–27, 2011.

[7] Google, “Google Noto Fonts,” 2014. [Online]. Available: https://www.google.com/get/noto/#sans-bali. [Accessed: 27-Jan-2017].

[8] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms (3rd ed.) Massachusetts Institute of Technology*. Boston: Massachusetts Institute of Technology, 2009.

[9] Author, 2018.

[10] Author, 2017.

[11] Made Suatjana, “Bali Simbar | Babad Bali,” 1999. [Online]. Available: http://www.babadbali.com/aksarabali/balisimbar.htm. [Accessed: 01-Apr-2017].

[12] Norbert Lindenberg, “Bringing Balinese to iOS,” 2015. [Online]. Available: http://norbertlindenberg.com/2015/10/bringing-balinese-to-ios/. [Accessed: 01-Apr-2017].

[13] I. D. A. M. Sartini, M. W. A. Kesiman, and I. G. M. Darmawiguna, “Pengembangan Text to Digital Image Converter Untuk Dokumen Aksara Bali,” in *Seminar Nasional Pendidikan Teknik Informartika (Senapati)*, 2012, pp. 280–296.

[14] Alit Jaya Trisna, “Aksara Bali - Android Apps on Google Play,” 2014. [Online]. Available: https://play.google.com/store/apps/details?id=com.skripsi.aksara. [Accessed: 05-Jul-2017].

[15] Agus Made, “Transliterasi Aksara Bali - Android Apps on Google Play,” 2016. [Online]. Available:

https://play.google.com/store/apps/details?id=com.agusmade.aksarabali. [Accessed: 05-Jul-2017].

[16] The Unicode Consortium, *The Unicode Standard, Version 5.0, Addison-Wesley Professional*. Addison-Wesley Professional, 2006.

[17] GitHub, “Add NEWS for license change - googlei18n/noto-fonts,” 2015. [Online]. Available: https://github.com/googlei18n/noto-fonts/commit/787e6b68b264428063342415d7c96afa3a32eec9. [Accessed: 21-Nov-2017].

[18] B. B. Narendra, “Pembuatan Smart Font Aksara Bali, Undergraduate Thesis,” Institut Teknologi Bandung, Bandung, 2008.

[19] I. W. Simpen, *Pasang Aksara Bali*. Denpasar: Upada Sastra, 1995.

[20] I. W. Simpen, *Purwa Aksara*. Denpasar: Upada Sastra, 2002.

[21] AutoIt, “AutoIt Scripting Language,” 2018. [Online]. Available: https://www.autoitscript.com/site/autoit/. [Accessed: 21-May-2018].

[22] Makeblock, “mDrawBot: A Four-in-One Drawing Robotic Kit,” 2013. [Online]. Available: https://github.com/Makeblock-official/mDrawBot.

[23] Ida Bagus Adi Sudewa, “The Balinese Alphabet, v0.6 | Babad Bali,” 2003. [Online]. Available: http://www.babadbali.com/aksarabali/alphabet.htm. [Accessed: 01-Apr-2017].

[24] ISO, “ISO/IEC International Standard ISO/IEC 10646: Information Technology - Universal Coded Character Set (UCS).” Switzerland, 2010.

[25] Author, 2017.

[26] Author, 2012.

[27] Author, 2014.

[28] Author, 2014.

[29] Author, 2014.

[30] I Nengah Tinggen, *Celah-Celah Kunci Pasang Aksara Bali, Rhika Dewata*. Singaraja: Rhika Dewata, 1994.

[31] DisBud-Bali, *Pedoman Pasang Aksara Bali, Dinas Kebudayaan Provinsi Bali*. Denpasar: Department of Culture, Bali Province, 1997.

[32] I. G. K. Anom *et al.*, *Kamus Bali - Indonesia Beraksara Latin dan Bali*. Denpasar: Badan Pembina Bahasa, Aksara, dan Sastra Bali, 2008.

[33] Author, 2018.