

Static Sign Language Recognition Using Deep Learning

Lean Karlo S. Tolentino, Ronnie O. Serfa Juan, August C. Thio-ac, Maria Abigail B. Pamahoy, Joni Rose R. Forteza, and Xavier Jet O. Garcia

Abstract—A system was developed that will serve as a learning tool for starters in sign language that involves hand detection. This system is based on a skin-color modeling technique, i.e., explicit skin-color space thresholding. The skin-color range is predetermined that will extract pixels (hand) from non-pixels (background). The images were fed into the model called the Convolutional Neural Network (CNN) for classification of images. Keras was used for training of images. Provided with proper lighting condition and a uniform background, the system acquired an average testing accuracy of 93.67%, of which 90.04% was attributed to ASL alphabet recognition, 93.44% for number recognition and 97.52% for static word recognition, thus surpassing that of other related studies. The approach is used for fast computation and is done in real time.

Index Terms—ASL alphabet recognition, sign language recognition, static gesture.

I. INTRODUCTION

Communication is essential in building a nation. Good communication leads to better understanding, and it encompasses all the members of the community, including the deaf. In the Philippines, 1.23% of the entire population is either deaf, mute or hearing impaired [1]. Sign language bridges the gap of communication with other people. However, most hearing people do not understand sign language and learning it is not an easy process. As a result, there is still an undeniable barrier between the hearing impaired and hearing majority.

Over the past few decades, many efforts have been made in creating a sign language recognition (SLR) system. There are two main categories in SLR, namely isolated sign language recognition and continuous sign classification. Zhang *et al.* and Wang *et al.* [2], [3] focus on isolated SLR, whereas Starner *et al.* and Vogler *et al.* [4], [5] pay attention to continuous SLR. The hidden Markov model (HMM) works on continuous SLR because HMM enables the segmentation of data stream into its continuous signs implicitly, thus bypassing the hard problem of segmentation entirely.

The manuscript received May 12, 2019; revised October 25, 2019. This work was supported in part by the University Research and Extension Council and the University Research and Development Services Office of the Technological University of the Philippines.

L. K. S. Tolentino is with the Department of Electronics Engineering, Technological University of the Philippines and the University Extension Services Office, Technological University of the Philippines, Philippines (e-mail: leankarlo_tolentino@tup.edu.ph).

R. O. Serfa Juan is with the Department of Electronic Engineering, Cheongju University, South Korea (e-mail: ronnieerfajuan@cju.edu.kr).

A. C. Thio-ac, M. A. B. Pamahoy, J. R. R. Forteza, and X. J. O. Garcia are with the Department of Electronics Engineering, Technological University of the Philippines, Philippines (e-mail: electronics@tup.edu.ph).

The SLR architecture can be categorized into two main classifications based on its input: data gloves-based and vision-based. Chouhan *et al.* [6] use smart gloves to acquire measurements such as the positions of hands, joints orientation, and velocity using microcontrollers and specific sensors, i.e., accelerometers, flex sensors, etc. There are other approaches to capturing signs by using motion sensors, such as electromyography (EMG) sensors, RGB cameras, Kinect sensors, leap motion controllers or their combinations [5], [7]–[9]. The advantage of this approach is having higher accuracy, and the weakness is that it has limited movement.

In recent years, the involvement of vision-based techniques has become more popular, of which input is from camera (web camera, stereo camera, or 3D camera). Sandjaja and Marcos [10] used color-coded gloves to make hand detection easier. A combination of both architectures is also possible, which is called the hybrid architecture [9]. While these are more affordable and less constraining than data gloves, the weakness of this approach is lower accuracy and high computing power consumption.

The architecture of these vision-based systems [11]–[14] is typically divided into two main parts. The first part is the feature extraction, which extracts the desired features from a video by using image processing techniques or the computer vision method. From the extracted and characterized features, the second part that is the recognizer should be learning of the pattern from training data and correct recognition of testing data on which machine algorithms were employed.

Most of the studies mentioned above focus on translating the signs typically made by the hearing-impaired person or the signer to word(s) that the hearing majority or non-signer can understand. Although these studies proved that technology is useful in so many ways, their proponents think that these are intrusive to some hearing-impaired individuals. Instead, the proponents proposed a system that will help those non-signers who want to learn basic static sign language and not being intrusive at the same time. It is also important to mention that there are applications implemented on mobile phones that help the non-signer to learn sign language through several videos installed on the apps. However, most of these apps require a large amount of storage and good internet connection.

The proposed study aims to develop a system that will recognize static sign gestures and convert them into corresponding words. A vision-based approach using a web camera is introduced to obtain the data from the signer and can be used offline. The purpose of creating the system is that it will serve as the learning tool for those who want to know more about the basics of sign language such as alphabets, numbers, and common static signs. The proponents provided a white background and a specific location for image processing of the hand, thus, improving the accuracy of the

system and used Convolutional Neural Network (CNN) as the recognizer of the system. The scope of the study includes basic static signs, numbers and ASL alphabets (A–Z). One of the main features of this study is the ability of the system to create words by fingerspelling without the use of sensors and other external technologies.

For the purpose of the study, some of the letters in ASL alphabets were modified. Fig. 1 presents the ASL alphabets that will be fed onto the system. It can be seen that letters e, m, and n were exaggerated compared to the original gesture, whereas j and z were converted to static gestures by getting only their last frame.

ASL also is strict when it comes to the angle of the hands while one is hand signing; again, for the purpose of the study, the angles of the hands for letters *p*, *x* and *t* were modified for their uniqueness, which would greatly affect the accuracy of the system.

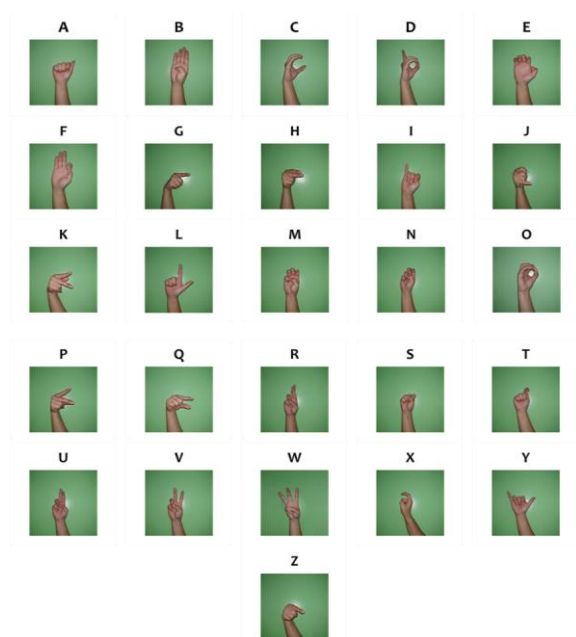


Fig. 1. ASL alphabet signs.

Fig. 2 presents a static gesture for each number provided. The system will be limited with numbers 1–10.

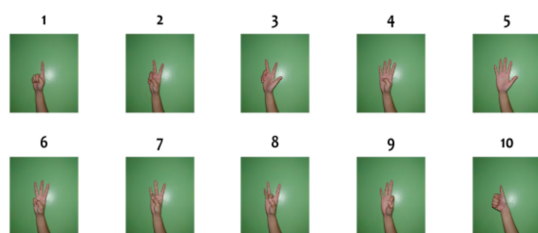


Fig. 2. ASL number signs.

In Fig. 3, static words are provided. Thirty-five words were chosen according to the results of the needs assessment survey conducted. The words were divided into four categories, namely family, communication, transportation, and common nouns or verbs.

II. REVIEW OF RELATED LITERATURE

For the past decades, research on SLR has been explored. Many studies used sensor-based devices such as *SignSpeak*.

This device used different sensors such as flex and contact sensors for finger and palm movements and accelerometers and gyros for the hand movement; then, by Principal Component Analysis, the gloves were trained to recognize different gestures, and each gesture was then classified into alphabets in real time. The device also used an Android phone to display the text and word received from the



Fig. 3. ASL static word signs.

gloves via Bluetooth. *SignSpeak* was found to have 92% accuracy [15]. There are other means of capturing signs by using motion sensors, such as electromyography (EMG) sensors [16], RGB cameras [17], Kinect sensors [18], and leap motion controller [19] or their combinations. Although these sensors provide accurate parameters in measurement of data, they also have limitations; first is their cost, as they require large-size datasets with diverse sign motion they going toned a high-end computers with powerful specifications; next is aesthetics, as the sensors are attached to the fingers and palms of a user, the user can encounter difficulties in setting up the device; ambient lighting conditions or backgrounds in real-world settings may also affect the recognition. Therefore, many researchers jumped from sensor-based to visual-based SLR.

Several methods have been developed in visual-based SLR. Because sign language includes static and dynamic movements, image, and video processing was explored by many.

Wang *et al.* [20] used color spaces to identify hand gestures and acquired segment images by setting a range of the skin color threshold. Hand gesture segmentation is simply done by using the hand skin threshold method. The system would not produce good results because of lighting conditions, skin color interference, and complex backgrounds that increased noise. There are three types of skin color

detection: the explicit range method, the nonparametric method and the parametric method [21]. The explicit range method differentiates the class of pixels into skin- and non-skin-based types from the assigned range of colors [22]. This technique is used mostly because of its non-complex approach and acceptable rate of computation. However, this technique is only limited for a generalized skin color scheme. Another approach was taken by Balbin *et al.* [23], who used colored gloves for the hands to be identified easily by setting an exact range of the hand skin color threshold (color of the gloves). To recognize the hand gesture, input images underwent various image processing methods or steps. First is pre-processing wherein images were converted into grayscale, and median filter is used to denoise the image. Next, is feature extraction wherein the color of the hand gloves was detected and isolated from the background. Then, the image had undergone pattern recognition. The system used Kohen self-organizing maps, which are the type of a neural network that can learn to identify patterns and group datasets in an unsupervised manner. The system was tested by five persons, and it achieved an accuracy of 97.6%.

These studies propose a complex yet manageable process of skin color thresholding; it can be seen that when only the bare hands of the signer are used, it is difficult for the system to recognize the gesture because of different hindrances such as noise. Other studies used colored gloves to solve the problem, whereas the present study proposed a system that can recognize static sign language without the aid of gloves or hand markings but still produce acceptable results.

III. METHODOLOGY

The system will be implemented through a desktop with a 1080P Full-HD web camera. The camera will capture the images of the hands that will be fed in the system. Note that the signer will adjust to the size of the frame so that the system will be able to capture the orientation of the signer's hand.

Fig. 4 illustrates the conceptual framework of the system. When the camera has already captured the gesture from the user, the system classifies the test sample and compares it in the stored gestures in a dictionary, and the corresponding output is displayed on the screen for the user.

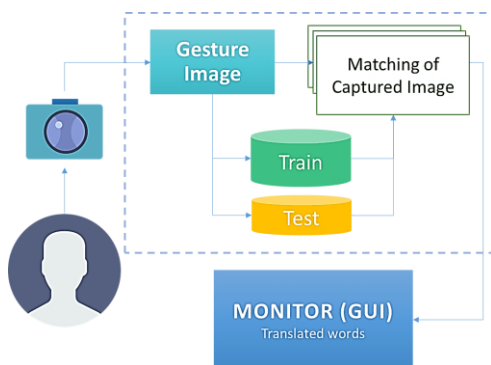


Fig. 4. Conceptual framework.

A. Gathering of Training Data, Image Augmentation, and Cropping Procedures

Gathering of datasets for static SLR was done through the use of continuous capturing of images using Python. Images

were automatically cropped and converted to a 50×50 pixels black and white sample. Each class contained 1,200 images that were then flipped horizontally, considering the left-handed signers. Fig. 5 presents the sample for flipped images. In Fig. 6, an example of capturing datasets is provided.



Fig. 5. Sample dataset of flipped images using ImageGenerator.

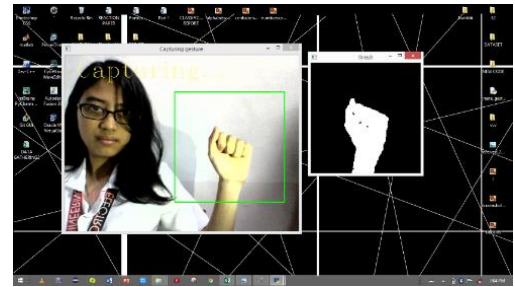


Fig. 6. Sample on how the datasets were being captured continuously using the vision-based technique.

B. Hand Skin Color Detection using Image Processing

For improved skin color recognition, the signer was advised to have a clear background for the hands, which will make it easier for the system to detect the skin colors. Skin detection took place by using `cv2.cvtColor`. Images were converted from RGB to HSV. Through the `cv2.inRange` function, the HSV frame was supplied, with the lower and upper ranges as the arguments. The mask was the output from the `cv2.inRange` function. White pixels in the mask produced were considered to be the region of the frame weighed as the skin. Although black pixels are disregarded, `cv2.erode` and `cv2.dilate` functions remove small regions that may represent a small false-positive skin region. Then, two iterations of erosions and dilations were done using this kernel. Lastly, the resulting masks were smoothened using a Gaussian blur.

C. Network Layers

The goal of this study is to design a network that can effectively classify an image of a static sign language gesture to its equivalent text by a CNN. To attain specific results, we used Keras and CNN architecture containing a set of different layers for processing of training of data.

The convolutional layer is composed of 16 filters, each of which has a 2×2 kernel. Then, a 2×2 pooling reduces spatial dimensions to 32×32 . From 16 filters of the convolutional layers, filters are increased to 32, whereas that of the Max Pooling filters is increased to 5×5 . Then, the number of filters in the CNN layers is increased to 64, but maxpooling is still at 5×5 . Dropout(0.2) functions with randomly disconnecting each node from the current layer into the next layer.

The model is now being flattened or is now converted into a vector; then, the dense layer is added. The fully connected layer is being specified by the dense layer along with rectified linear activation.

We finished the model with the SoftMax classifier that would give the predicted probabilities for each class label.

The network's output is now a 25-dimension vector similar to each sign language alphabets, a 10-dimension vector corresponding to sign language numbers. Then, for a 35-dimension vector corresponding to static sign language gestures, each class was trained through the network individually.

D. Training the System

The training for character and SSL recognition was done separately; each dataset was divided into two: training and testing. This was done to see the performance of the algorithm used. The network was implemented and trained through Keras and TensorFlow as its backend using a Graphics Processing Unit GT-1030 GPU.

The network uses a Stochastic gradient descent optimizer as its optimizer to train the network having a learning rate of 1×10^{-2} . The total number of epochs used to train the network is 50 epochs with a batch size of 500. The images were resized to (50, 50, 1) for training and testing.

We use a stochastic gradient descent optimizer, also known as the incremental gradient descent, to minimize the batch size of large datasets.

The batch gradient descent performs redundant computations for large datasets as gradients are recalculated before each parameter update for similar examples. By performing one update at a time, SGD eliminates this redundancy. It is typically much faster and can also be used for online learning.

IV. TESTING

The project was tested by 30 individuals: 6 were sign language interpreters, and 24 were students with and without knowledge in using sign language. Thirty samples were preferred to be able to use Student's t-test or the significance validation of the study and to prove the reliability of the system in recognizing static hand gestures from the hands of people who were not in the dataset

Three trials were conducted in each letter/number/word gesture recognition. Each trial has a duration of 15 seconds per letter/number/word gesture recognition. If the system did not print the equivalent text of the signs within the allotted time, the output was considered to be incorrect.

A. Testing Procedure

Before the actual recognition of the signs, the user must calibrate the light to ensure that the skin masking of the hand is detected and has less noise; the calibration can be done by moving the lampshade sideways. It is recommended that the light is not directly hitting the hand. The system is sensitive to light; thus, determining the proper place of the lamp should be considered. If the edges of the hand in the masking are detected clearly, the user may begin to use the translator.

For the signs to be recognized, the hand should in front of the camera. The detection can only be done if the hand is inside the box that can be seen on the screen of a computer's monitor. Since the size of the hand of each individual is different, a user may move his/her hand back and forth to fit inside the virtual box. The user should then wait for the system to generate the desired equivalent of the signs in textual form. It also recommended that the user's hand does

not make any movement until the system generates the output.

To know the rate of learning, the researcher measures the time of producing the translated static signs using a stopwatch and repeats this three times.

B. Testing of the Accuracy Formula

To verify the accuracy of the letter/number/word gestures recognition, the number of the correctly recognized letters/words/numbers that appeared on the screen was added and divided by the product of the total number of users multiplied by the number of trials.

The correct recognition is acquired when the signs made by the user are translated and their respective equivalents are produced in textual form within the duration of 15 seconds. If the system generates the equivalent word/letter/number beyond 15 seconds, it is not included in the total number of correct recognized letters/words/numbers.

$$\text{Accuracy rate} = \frac{\text{Total nos. of correct recognized letters from users}}{(\text{Total no. of users})(\text{No. of Trails})} \quad (1)$$

V. RESULTS AND DISCUSSIONS

In a recognition system, the accuracy rate of the recognition is the greatest concern. Table 1 presents the accuracy rate and average recognition time of each letter from all the trials. Thirty users tested the system with three trials.

TABLE I: LETTER RECOGNITION ACCURACY

Letter	Correctly Recognized Gestures	Incorrect Recognized Gestures	Accuracy (%)	Ave. Time (s)
A	90	0	100	2.02
B	82	8	91.11	4.01
C	90	0	100	2.2
D	90	0	100	2.46
E	87	3	96.67	3.59
F	84	6	93.33	4.95
G	89	1	98.89	2.85
H	76	14	84.44	5.39
I	85	5	94.44	3.37
J	81	9	90	4.57
K	85	5	94.44	4.06
L	89	1	98.89	2.38
M	74	16	82.22	5.98
N	81	9	90	3.92
O	81	9	90	3.62
P	78	12	86.67	4.97
Q	86	4	95.56	3.76
R	71	19	78.89	6.24
S	78	12	86.67	4.6
T	67	23	74.44	7.62
U	77	13	85.56	5.11
V	76	14	84.44	5.07
W	87	3	96.67	2.46
X	78	12	86.67	4.98
Y	84	6	93.33	3.5
Z	81	9	67.78	8.31
Overall Rating			90.04%	4.31

The accuracy rate of each letter was obtained using formula (1), wherein the total number of the correctly recognized words from all the users from all the trials was totaled and divided by the total number of samples, which is the number of users (30) multiplied by the number of trials (3).

It can be seen that the unique letters such as A, C and D got the highest accuracy with 100% rating, and the lowest was Z with 67.78%.

The overall letter recognition accuracy of the system was achieved by getting the average of each letter's accuracy. The system attained 90.04% accuracy. It also attained an average time of 4.31 second real-time letter recognition of hands that were not in the dataset. This was obtained by getting the average of the recognition time of each letter.

The same computation was performed in number recognition and static word recognition.

Table II presents the accuracy result of each number from all the trials. It can be seen that number 5 got the highest accuracy with 100% rating, and the lowest was 8 with 83.33%. The number recognition attained 93.44% accuracy with an average time of 3.93 second real-time number recognition of hands that were not in the dataset.

TABLE II: NUMBER RECOGNITION ACCURACY

Number	Correctly Recognized Gestures	Incorrect Recognized Gestures	Accuracy (%)	Ave. Time (s)
1	89	1	98.89	2.28
2	89	1	98.89	2.52
3	75	15	83.33	5.56
4	84	6	93.33	4.36
5	90	0	100	2.4
6	82	8	91.11	5.57
7	79	11	87.78	5.22
8	75	15	83.33	5.69
9	89	1	98.89	2.82
10	89	1	98.89	2.91
Overall Rating			93.44	3.93

TABLE III: STATIC WORD RECOGNITION ACCURACY

Static Word	Correctly Recognized Gestures	Incorrect Recognized Gestures	Accuracy (%)	Ave. Time (s)
Airplane	83	7	92.22	4
Bus	87	3	96.67	3.56
Calm down	90	0	100	2.33
Car	88	2	97.78	3.07
Church	87	3	96.67	3.21
Cover	89	1	98.89	3.41
Family	90	0	100	2.58
Father	83	7	92.22	3.29
Fine	87	3	96.67	2.87
Hello	89	1	98.89	2.73
Help	86	4	95.56	3.06
Home	90	0	100	2.33
Hungry	88	2	97.78	2.85
I am/ I'm	87	3	96.67	2.97
I hate you	88	2	97.78	2.81
I love you	88	2	97.78	2.58
Key	89	1	98.89	3.09
Lock	90	0	100	2.64
Love	90	0	100	2.56
Money	89	1	98.89	2.54
Mother	88	2	97.78	2.83
No	84	6	93.33	3.32
Okay	90	0	100	2.19
Pray	90	0	100	2.22
Ship	88	2	97.78	3.18
Sorry	87	3	96.67	3.09
Stand	88	2	97.78	2.88
Taxi	84	6	93.33	4.24
Telephone	84	6	93.33	3.4
Time	88	2	97.78	3.12
Water	87	3	96.67	2.67
Where	90	0	100	2.44
Why	88	2	97.78	2.62
Yes	89	1	98.89	2.52
Overall Rating			90.04%	4.31

Table III presents the accuracy result of each static word. It can be seen that words "calm down," "family," "home," "love," etc. got the highest accuracy with 100% rating, and the lowest was the word "father" with 92.22%. Thus, the more unique the gesture, the better accuracy it can get. The static word recognition attained 97.52% accuracy with an average time of 2.9 second real-time number recognition of hands that were not in the dataset.

Among the three systems tested, the recognition for static word gesture got the highest accuracy and average time of 97.52% and 2.9 sec, respectively. The dataset of the static word gestures indicates that the gestures are more unique from each other compared to the ASL letter and alphabet gestures.

To get the final accuracy of the static SLR system, the accuracies obtained from the letter recognition, static word recognition, and number recognition were totaled and averaged, thus giving an accuracy of 93.667%.

For the learning accuracy of the system, Table IV is the summary of the average time for the letter, number, and static word recognition system in each trial.

It can be seen that the average recognition time from trial 1 to trial 3 decreases, thus validating the learning accuracy of the system. To get the overall average time of the system in each trial, the average times of the letter, number, and static word recognition system per trial were also averaged. This results to a better comparison for the recognition time. From 5.21 s recognition time in trial 1, the time went down to 2.66 s in trial 3.

TABLE IV: SUMMARY OF THE AVERAGE TIME IN EACH TRIAL

	TRIAL 1	TRIAL 2	TRIAL 3
Letters	6.31	3.75	2.86
Numbers	5.68	3.32	2.79
Words	3.63	2.75	2.33
Ave. time per trial of the whole system (s)	5.21	3.27	2.66

To further verify the learning accuracy of the static sign language and character recognition, the acquired average time in each trial was compared using the Student's t-test: paired two-tailed samples for mean with a 0.05 significance level. The null hypothesis is that the means of the groups are equal. Trials 1 and 2 were first compared; then, trials 2 and 3 followed. Table V presents the summary of the Student's t-test results.

TABLE V: SUMMARY OF STUDENT'S T-TEST RESULTS

		T1 vs. T2	T2 vs. T3
Letter	t Statistic	9.8247	6.9388
	$P(T \leq t)$, two-tailed	4.58×10^{-10}	2.84×10^{-7}
	t Critical, two-tailed	2.0595	2.0595
Number	t Statistic	5.0622	5.5812
	$P(T \leq t)$, two-tailed	0.0007	0.0003
	t Critical, two-tailed	2.2622	2.2622
Word	t Statistic	9.6317	4.4453
	$P(T \leq t)$, two-tailed	3.02×10^{-11}	0.0001
	t Critical, two-tailed	2.0322	2.0322

To fully reject the null hypothesis, both p and t values were compared. Since the p-values of the samples were less than the significance level (0.05), the null hypothesis can be rejected. For the t value, the t statistic should be greater than t critical. Table V indicates that T1 vs. T2 and T2 vs. T3 obtained a greater t statistic value than the t critical value; therefore, we can say that there is a significant difference between the means of each trial; this is strong enough evidence for us to fully reject the null hypothesis.

To validate the effectiveness of the system, Table VI presents the summary of the evaluation result in terms of functionality, reliability, usability, efficiency, and learning impact of the SSL recognition system.

The system was evaluated by 50 persons using Likert's scale with poor, fair, average, good, and excellent as the rating scale; poor = 1, average = 2, fair = 3, good = 4 and excellent = 5.

Here, there were three questions concerning functionality, two questions for reliability, four questions for usability, and one question each for efficiency and learning impact. The total score was computed by adding all the ratings of all the users in questions concerning each criterion. The goal score was the total score when all the users rated each criterion as excellent. The system was able to achieve 88.46% rating.

TABLE VI: SUMMARY OF EVALUATION RESULTS

	No. of evaluator	No. of questions	Total Score	Goal Score	Approval Percentage
Functionality	50	3	642	750	85.86
Reliability	50	2	433	500	86.86
Usability	50	4	901	1,000	90.1
Efficiency	50	1	215	250	86
Learning Impact	50	1	235	250	94
Total score:					88.46

It can be noticed that the learning impact of the system to the users and its usability acquired an admirable result; hence, the system can be helpful in learning static sign language.

VI. CONCLUSION

The main objective of the project was to develop a system that can translate static sign language into its corresponding word equivalent that includes letters, numbers, and basic static signs to familiarize the users with the fundamentals of sign language. The researchers created an assessment plan, and a series of tests were done to guarantee the significance of the functionalities of the system intended for non-signers. Results of testing were remarkable marks in terms of usability and learning impact of the system. This project was done with proper aid and consultation from sign language experts.

Reaching the training phase of the development, one of the objectives of the study was to match or even exceed the accuracy of the studies presented using deep learning. Our system was able to achieve 99% training accuracy, with testing accuracy of 90.04% in letter recognition, 93.44% in number recognition and 97.52% in static word recognition, obtaining an average of 93.667% based on the gesture recognition with limited time. Each system was trained using 2,400, 50 × 50 images of each letter/number/word gesture.

In comparison to other systems which only recognized ASL alphabets, our study added more gestures, making SLR more useful and effective. In the literature [24]–[27], the systems only recognized ASL alphabets, whereas in our system we included numbers; in a study by Balbin *et al.* [23], the system only recognized five Filipino words and used colored gloves for hand position recognition, thus having the best accuracy.

Despite it having average accuracy, our system is still well-matched with the existing systems, given that it can

perform recognition at the given accuracy with larger vocabularies and without an aid such as gloves or hand markings.

ACKNOWLEDGMENT

The researchers would like to thank Rev. Fr. Luke Moortgat, Ms. Donna, Ms. Sarah, Ms. Tess, Ms. Carol, and the whole Precious Deaf Community for the endless support throughout the study.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

L.K.S.T., R.O.S.J., A.C.T., J.R.R.F., and X.J.O.G. conducted the research; L.K.S.T., A.C.T., M.A.B.B., J.R.R.F., and X.J.O.G. analyzed the data; L.K.S.T. and M.A.B.B. wrote the paper; L.K.S.T. and R.O.S.J. edited and reviewed the paper; all authors had approved the final version.

REFERENCES

- [1] F. R. Session, A. Pacific, and S. Africa, *Senat '2117*, 2014, pp. 1–3.
- [2] L. G. Zhang, Y. Chen, G. Fang, X. Chen, and W. Gao, "A vision-based sign language recognition system using tied-mixture density HMM," in *Proc. the 6th International Conference on Multimodal Interfaces*, 2004, pp. 198–204.
- [3] Q. Wang, X. Chen, L. G. Zhang, C. Wang, and W. Gao, "Viewpoint invariant sign language recognition," *Computer Vision and Image Understanding*, vol. 108, no. 1–2, pp. 87–97, 2007.
- [4] T. Stamer, J. Weaver, and A. Pentland, "Real-time american sign language recognition using desk and wearable computer based video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 12, pp. 1371–1375, 1998.
- [5] C. Vogler and D. Metaxas, "Handshapes and movements: Multiple-channel american sign language handshapes and movements: Multiple-channel ASL recognition," in *Proc. International Gesture Workshop*, 2003, pp. 247–258.
- [6] T. Chouhan, A. Panse, A. K. Voona, and S. M. Sameer, "Smart glove with gesture recognition ability for the hearing and speech impaired," in *Proc. 2014 IEEE Global Humanitarian Technology Conference-South Asia Satellite (GHTC-SAS)*, 2014, pp. 105–110.
- [7] M. Süzgün, H. Özdemir, N. Camgöz, A. Kindiroğlu, D. Başaran, C. Togay, and L. Akarun, "HospiSign: An interactive sign language platform for hearing impaired," *Journal of Naval Sciences and Engineering*, vol. 11, no. 3, pp. 75–92, 2015.
- [8] J. A. Deja, P. Arceo, D. G. David, P. Lawrence, and R. C. Roque, "MyoSL: A Framework for measuring usability of two-arm gestural electromyography for sign language," in *Proc. International Conference on Universal Access in Human-Computer Interaction*, 2018, pp. 146–159.
- [9] C. Ong, I. Lim, J. Lu, C. Ng, and T. Ong, "Sign-language recognition through Gesture & Movement Analysis (SIGMA)," *Mechatronics and Machine Vision in Practice*, vol. 3, pp. 232–245, 2018.
- [10] I. N. Sandjaja and N. Marcos, "Sign language number recognition," in *Proc. 2009 Fifth International Joint Conference on INC, IMS and IDC*, 2009, pp. 1503–1508.
- [11] E. P. Cabalfin, L. B. Martinez, R. C. L. Guevara, and P. C. Naval, "Filipino sign language recognition using manifold learning," in *Proc. TENCON 2012 IEEE Region 10 Conference*, 2012, pp. 1–5.
- [12] P. Mekala, Y. Gao, J. Fan, and A. Davari, "Real-time sign language recognition based on neural network architecture," in *Proc. 2011 IEEE 43rd Southeastern Symposium on System Theory*, 2011, pp. 195–199.
- [13] J. P. Rivera and C. Ong, "Recognizing non-manual signals in filipino sign language," in *Proc. Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, 2018, pp. 1–8.
- [14] J. P. Rivera and C. Ong, "Facial expression recognition in filipino sign language: Classification using 3D Animation units," in *Proc. the 18th Philippine Computing Science Congress (PCSC 2018)*, 2018, pp. 1–8.
- [15] J. Bukhari, M. Rehman, S. I. Malik, A. M. Kamboh, and A. Salman, "American sign language translation through sensory glove;

SignSpeak,” *International Journal of u-and e-Service, Science and Technology*, vol. 8, no. 1, pp. 131–142, 2015.

- [16] J. Wu, Z. Tian, L. Sun, L. Estevez, and R. Jafari, “Real-time American sign language recognition using wrist-worn motion and surface EMG sensors,” in *Proc. 2015 IEEE 12th International Conference on Wearable and Implantable Body Sensor Networks (BSN)*, 2015, pp. 1–6.
- [17] D. Mart, *Sign Language Translator Using Microsoft Kinect XBOX 360 TM*, 2012, pp. 1-76.
- [18] C. Dong, M. C. Leu, and Z. Yin, “American sign language alphabet recognition using microsoft kinect,” in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015, pp. 44-52.
- [19] B. Toghiani-rizi, C. Lind, M. Svensson, and M. Windmark, “Static gesture recognition using leap motion,” *arXiv preprint arXiv:1705.05884*, 2017.
- [20] H. Wang, Y. Wang, and Y. Cao, “Video-based face recognition: A survey,” *World Academy of Science, Engineering and Technology*, vol. 60, pp. 293-302, 2009.
- [21] F. Gianfelici, C. Turchetti, and P. Crippa, “A non-probabilistic recognizer of stochastic signals based on KLT,” *Signal Processing*, vol. 89, no. 4, pp. 422–437, 2009.
- [22] M. K. Ahuja and A. Singh, “Hand gesture recognition using PCA,” *International Journal of Computer Science Engineering & Technology*, vol. 5, no. 7, pp. 267–271, 2015.
- [23] J. R. Balbin, D. A. Padilla, F. S. Caluyo, J. C. Fausto, C. C. Hortinela, C. O. Manlises, C. K. S. Bernardino, E. G. Finones, and L. T. Ventura, “Sign language word translator using neural networks for the aurally impaired as a tool for communication,” in *Proc. 2016 6th IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, 2016, pp. 425-429.
- [24] S. Shahriar, A. Siddiquee, T. Islam, A. Ghosh, R. Chakraborty, A. I. Khan, C. Shahnaz, and S. A. Fattah, “Real-time american sign language recognition using skin segmentation and image category classification with convolutional neural network and deep learning,” in *Proc. TENCON 2018-2018 IEEE Region 10 Conference*, 2018, pp. 1168-1171.
- [25] R. Daroya, D. Peralta, and P. Naval, “Alphabet sign language image classification using deep learning,” in *Proc. TENCON 2018-2018 IEEE Region 10 Conference*, 2018, pp. 646-650.
- [26] T. Kim, G. Shakhnarovich, and K. Livescu, “Finger-spelling recognition with semi-markov conditional random fields,” in *Proc. 2013 IEEE International Conference on Computer Vision*, 2013, pp. 1521–1528.
- [27] N. Pugeault and R. Bowden, “Spelling it out: Real-time asl finger-spelling recognition,” in *Proc. 2011 IEEE International Conference on Computer Vision Workshop*, 2011, pp. 1114–1119.



Lean Karlo S. Tolentino received his B.S. in electronics and communications engineering from the Technological University of the Philippines (TUP), Manila and M.S. in electronics engineering major in microelectronics from Mapua University, in 2010 and 2015, respectively. He was the head of the Department of Electronics Engineering, TUP Manila from 2017 to 2019. He is currently the director of the University Extension Services Office, TUP since 2019. He is a member of IECEP and IEEE. His research interests include microelectronics and information and computing technologies



frequency identification (RFID), advanced driver assistance system (ADAS) technology and Controller Area Network (CAN).



systems and information and computing technologies.



Maria Abigail B. Pamahoy graduated as the class salutatorian in elementary and pursued high school in a science class at Sapang Palay National High School. She received her B.S. in electronics engineering at Technological University of the Philippines, Manila last 2019. Ms. Pamahoy is a member of IECEP-MSC and OECEs and an active member of Assumption Development Foundation where she developed her passion on service, leadership and excellence.



Joni Rose R. Forteza finished her studies in elementary and high school at Marilao Central School and AFG. Bernardino Memorial Trade School, respectively. She was an active member of Philippine Girl Scout and Drum and Lyre Band during her elementary and high school days. She is a graduate of B.S. in electronics engineering major in microelectronics at Technological University of the Philippines, Manila last 2019.



Xavier Jet O. Garcia finished electronics engineering technology, a 3-year vocational course, at the Technological University of the Philippines, Manila last 2015. He also finished his B.S. in electronics engineering major in microelectronics at Technological University of the Philippines, Manila last 2019.