Malaysian Sign Language Recognition Using 3D Hand Pose Estimation

Kavishaalini Padmanand
Faculty of Computer Science and Information Technology
Universiti Malaysia Sarawak
Sarawak, Malaysia
kavishaa.pk@gmail.com

Phei-Chin Lim
Faculty of Computer Science and Information Technology
Universiti Malaysia Sarawak
Sarawak, Malaysia
pclim@unimas.my

2022 International Conference on Digital Transformation and Intelligence (ICDI) | 979-8-3503-9700-0/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICDI57181.2022.10007170

Abstract—**Sign languages are one of those mediums for hearing-impaired people. These languages transmit meaning by visual-manual treatment, or more simply, hand movement. Currently, there are only 95 sign language interpreters registered with the Malaysian Federation of the Deaf as of 2020, compared to 40,389 hearing-impaired individuals with disabilities registered with the welfare department which is a problem. Therefore, with the use of deep-learning technology, this paper proposes to alleviate the scarcity of Malaysian Sign Language interpreters for the benefit of hearing-impaired persons. The paper aims to test and report a sequenced 3D keypoint hand pose estimation model for Malaysian Sign Language Recognition and evaluate the implementation of action model in decoding basic poses of Malaysian Sign Language. According to the findings, the detecting of 3D keypoints and incorporating into LSTM models using deep learning machine learning platform and framework like TensorFlow and MediaPipe enables the detection of Malaysian sign language 3D hand posture estimation. The results demonstrated that 3D hand posture estimation may be utilised to estimate sign language in real time, providing for a better interpretation approach for the deaf community.**

Keywords—Deep learning, Sign Language, 3D hand pose estimation

I. INTRODUCTION

Communication is undeniably essential because it is believed to be the key to transferring information. It is known that communication necessitates the use of a medium for both the sender and the receiver to get access to any type of information. Sign languages (also known as signed languages) are one of those mediums for those with hearing-impairment. These languages transmit meaning by visual-manual treatment, or more simply, hand movement. However, sign languages were never considered legitimate until William Stokoe found in 1960, while observing deaf students chatting at Gallaudet College, that the American Sign Language (ASL) had its own grammatical structure [1]. As a result of continual social interactions, sign languages develop into sophisticated languages.

Malaysia contains three sign languages, according to the Ethnologue, a comprehensive inventory of the world's languages: Bahasa Isyarat Malaysia (BIM) or Kod Tangan Bahasa Malaysia (KTBM), Penang Sign Language, and Kuala Lumpur Sign Language (KLSL). Mr. Tan Yap, known as the "Father of the Deaf" began studying sign language interpreting in 1964 at Gallaudet College in Washington, D.C., USA [1] and established KLSL from ASL. He taught KLSL to deaf individuals in Johor who had not finished secondary school. In 1968, he founded a deaf school in Kuala Lumpur to educate deaf children [2]. BIM is a sign language that combines ASL with certain local signs [3] and has been used as the primary communication method among Malaysia's deaf community with the establishment of Malaysian Federation of the Deaf in 1998. Based on Malay basic words, BIM sign words were created for educational purposes [3]. Deaf youngsters learn sign language for Malay root words and affixes, which they utilize to express themselves in phrases according to Bahasa Malaysia grammar. In terms of syntax and lexicon, Bahasa Malaysia influenced BIM development [1]. Even though ASL has had a substantial influence on BIM, the two are unique enough to be considered independent languages.

It is apparent that sign language users have adapted regular lifestyles by reading lips and comprehending people. However, the opposite is not applicable. There are only 95 sign language interpreters registered with the Malaysian Federation of the Deaf as of 2020, compared to 40,389 hearing-impaired individuals with disabilities registered with the welfare department [4]. This makes it much more difficult for those with hearing impairment to interact with others as it is heavily dependent on whether an interpreter can be available to interpret.

II. RELATED WORKS

This section examined and discussed works related to recent methods in sign language recognition.

A. Sign Language Recognition Methods

Convolutional Neural Network (CNN) is a type of deep, feed-forward artificial neural network in machine learning. CNN typically includes of an input and output layer, as well as numerous hidden layers such as convolutional, pooling, fully connected, and normalising layers [5]. According to [6], the YOLO system uses CNN to recognise objects in real-time. The project focuses on the translation of Malaysian Sign Language hand gesture movements, which include the alphabet and fingerspelling. 2D CNN is utilised in the study by [7] to extract features from gestured alphabets and classify them into 24 alphabets.

When it comes to dealing with data sequences where the temporal dynamics connecting them are critical, recurrent neural networks (RNN) are the networks of choice [8]. RNN generates and recognises characters based on their past interactions [5]. There are two primary techniques to using RNN in sign language recognition: the Hidden Markov Model (HMM) and the Long Short-Term Memory (LSTM) architecture. [9] developed a method for extracting features and classifying the Vietnamese Sign Language's continuous dynamic motions (VSL). The data was collected using a Microsoft Kinect depth sensor. In another study, [5] introduce a modified four gated LSTM cell with 2D CNN for sign sentence recognition in another study.

Sensor modules are designed to record information about hand movement in the form of an electric signal (analogue voltage) [10]. [11] explored which algorithm combinations, configured with different parameters, and utilised with a sensor device, provide higher Arabic sign language (ArSL) recognition accuracy results in a gesture recognition system. On the other hand, [12] proposed a Data Glove-based system. This device, which is made up of flex sensors, presser sensors, and inertial measurement units that measure the motion of the fingers and wrists, is used to collect data on various gestures.

B. Hand Pose Estimation

Hand localization and pose estimation have grown in importance in the disciplines of human computer interface and computer vision in recent years [13]. Because hand expressions reflect many of the characteristics of human behaviour daily, hand posture estimations are critical for many human-computer interactions, such as augmented reality, virtual reality [14], and computer vision applications that need gesture tracking [15]. According to [16], hand pose estimations traditionally struggle with a wide range of pose articulations and occlusions, including self-occlusions. [17] state that the hand is one of the most mechanically and anatomically complicated elements of the human body.

To fulfil the hand morphological constraints, a 3D hand model is constructed that serves as a generative paradigm. To depict the hand, [18] used a hand model made up of ellipsoids, cuboids, cylinders, and cones while [19] utilized an ICP-PSO method to select the hand model parameters that were closest to the observed data. Deep learning-based discriminative techniques have recently gotten a lot of attention in the scientific community. This method demands the creation of a delicate 3D hand model as well as live iterative optimization [17]. [20] were among the first to apply a deep learning framework to the challenge of estimating hand posture on monocular images [21]. They also claim that discriminative ones are more likely to learn a regression function from training data, connecting the appearance of depth pictures to hand pose.

The most noteworthy hybrid approaches seek to blend discriminative and generative principles while preserving the benefits of each [22]. [23] combined RDF-based component labelling and a Gaussian mixture representation of depth into an objective function to estimate the best fit to the observed data.

While hand sign recognition models have advanced rapidly in recent years, there are still some issues that need to be addressed. As can be seen, machine learning algorithms play a significant role in detection and recognition. When it comes to dealing with data sequences where the temporal dynamics connecting them are critical, RNN are the networks of choice particularly LSTM approach. This project adapted the 3D hand pose estimation task, which is formulated as a nonlinear regression problem in which the mapping from input depth data to output joint coordinates is learned directly.

III. Materials And Methods

This project firstly captured a sequences of 3D keypoints for the training of a hand pose estimation model for the recognition of BIM before evaluating the recognition accuracy. Fig. 1 illustrated the framework with the three main phases in this project which are data acquisition, data pre-processing and classification. Tools and libraries used are TensorFlow, OpenCV and MediaPipe. Recognition models training and testing are conducted using a 64-bit operating system running 1.60GHz CPU with 4GB of RAM.

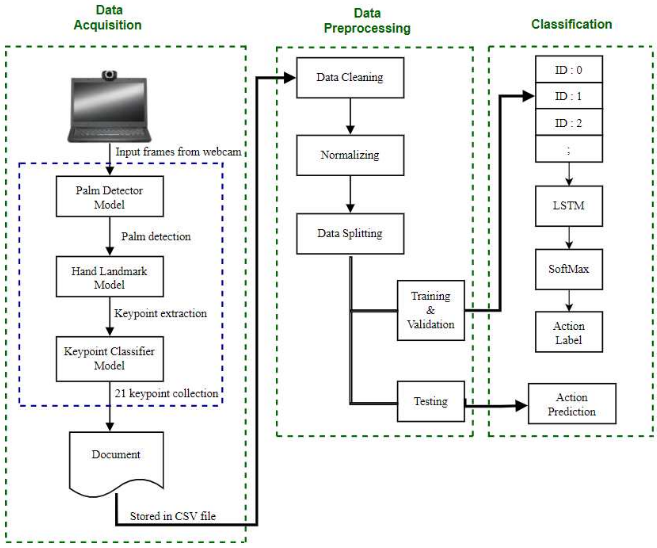


Fig. 1. The framework.

A. Data Acquisition

Data collected are the sign language actions; '*Terima Kasih', 'Apa Khabar', 'Khabar Baik', 'Sila', 'Minta Maaf', 'Jumpa Lagi'* and the alphabets from A to Z. Each actions will be demonstrated by 5 volunteers from the Malaysian Federation of the Deaf repeatedly 3 times. Sign actions data capturing is through webcam and accomplished using OpenCV library by configuring a video capture and then looping through every single frame while MediaPipe holistic and drawing functions are used to extract the frames and draw the keypoints on each frame. MediaPipe Hands [24] consist of a palm detection model that works on the full image and returns an oriented hand bounding box, and a hand landmark model that operates on the cropped image region defined by the palm detector and returns high-fidelity 3D hand keypoints. Fig. 2 depicts the detection of 21 landmark points by the hand landmark model. A simultaneous elimination task is carried out while extracting the landmark points. In this case, only the x- and y-coordinates detected by the hand landmark model are used for training the sign language recognition model.

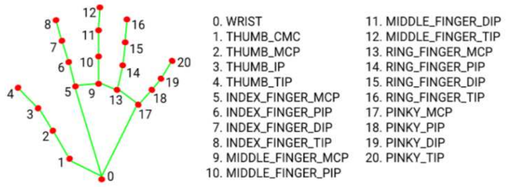
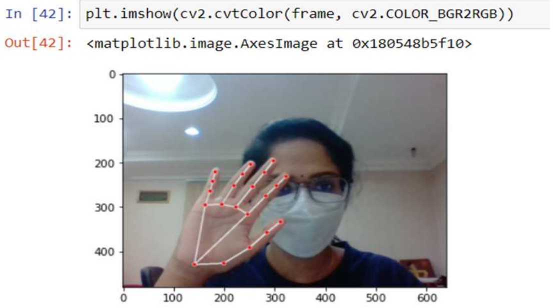


Fig. 2. Hand Coordinates [24].

Fig. 3 shows how OpenCV uses matplotlib function to help visualise the frame. The red dots are the landmarks while the white lines represent the connection.



```
In [42]: plt.imshow(cv2.cvtColor(frame, cv2.COLOR_BGR2RGB))
Out[42]: <matplotlib.image.AxesImage at 0x180548b5f10>
```

Fig. 3. Visualization of landmarks on the right hand

Fig. 4 showed some sample frames collected. The keypoints captured are concatenated into numpy arrays.

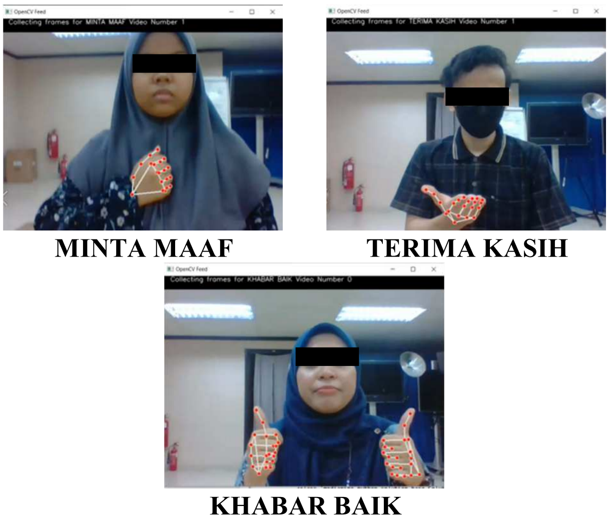
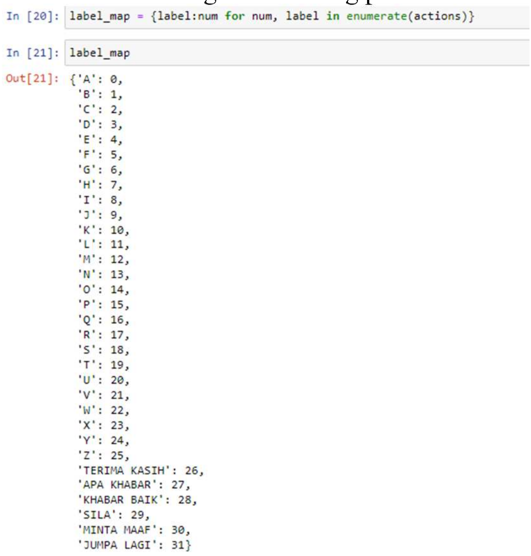

[redacted]
[redacted]
[redacted]
OpenCV Feed
Collecting frames for MINTA MAAF Video Number 1
OpenCV Feed
Collecting frames for TERIMA KASIH Video Number 1
MINTA MAAF
TERIMA KASIH
OpenCV Feed
Collecting frames for KHABAR BAIK Video Number 0
KHABAR BAIK

Fig. 4. Sample frames from data collection

B. Data Pre-processing

From data acquisition phase, only the keypoint's x- and y-coordinates extracted from each sequence of the sign action is collected and stored as numpy arrays. Models in MediaPipe Hands [24] may fail to detect the hand due to blurry images, resulting in a null entry in the numpy arrays. As a result, data cleaning is necessary or else the predictive model will be biased. Using indexes, rows containing null entries are removed from the numpy arrays.

To carry on with the data pre-processing and labelling, additional dependencies such as Scikit-learn and Keras were imported. A label array also known as a label dictionary is created to represent each one of the different actions captured (refer Fig. 5). Using the categorical function from tensorflow.keras, the initial labels are converted and passed through the label map. The train_test_split from Scikit-learn is used to create a training and a testing partition.

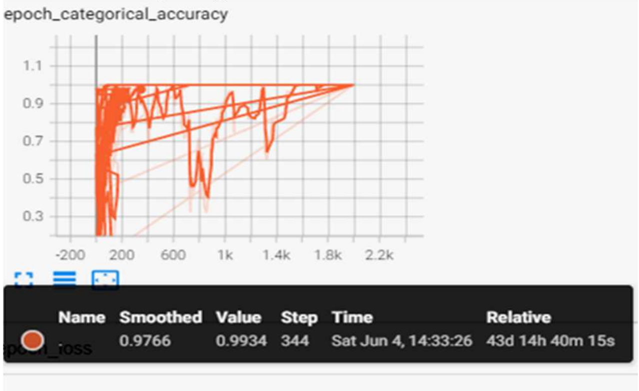


```
In [20]: label_map = {label:num for num, label in enumerate(actions)}

In [21]: label_map

Out[21]: {'A': 0,
          'B': 1,
          'C': 2,
          'D': 3,
          'E': 4,
          'F': 5,
          'G': 6,
          'H': 7,
          'I': 8,
          'J': 9,
          'K': 10,
          'L': 11,
          'M': 12,
          'N': 13,
          'O': 14,
          'P': 15,
          'Q': 16,
          'R': 17,
          'S': 18,
          'T': 19,
          'U': 20,
          'V': 21,
          'W': 22,
          'X': 23,
          'Y': 24,
          'Z': 25,
          'TERIMA KASIH': 26,
          'APA KHABAR': 27,
          'KHABAR BAIK': 28,
          'SILA': 29,
          'MINTA MAAF': 30,
          'JUMPA LAGI': 31}
```

Fig. 5. Label dictionary

C. Classification

Once the training and testing partition has been set, model with LSTM layers need to be trained. The LSTM layer will give a temporal component to building the neural network and allows it to perform the sign language estimation. This is done by using TensorFlow, an open-source library for machine learning. A sequential model is created using the Sequential API before adding 2-LSTM layers. The specified activation function used is Rectified Linear Unit (ReLU) as it is easier to use as it is a piecewise linear function of the input [25]. Adaptive Moment Estimation (Adam) optimiser performs well with SoftMax activation and requires less resources and makes the model converge faster [26], which can accelerate the learning speed and improve the effect. The loss function is set to categorical_crossentropy and metric used to evaluate the model performance is set to categorical_accuracy. The model is trained with epoch set to 2000. TensorBoard is used to monitor and check the training experiments. Fig. 6 showed performance of the training with 0.9934 accuracy at epoch 346/2000.



	Name	Smoothed	Value	Step	Time	Relative
●	poch_loss	0.9766	0.9934	344	Sat Jun 4, 14:33:26	43d 14h 40m 15s

Fig. 6. Data training performance graph

IV. Results And Discussion

This section discussed the experiments to evaluate the behaviour and performance of the proposed approach. Table I showed sample actions tested by 5 sign language users (members of MDF who are partially/fully deaf) and 5 non-sign language users (never used sign language in their daily life). Experiments was done to check if the tested model can perform correct recognition if the signs were not demonstrated properly or with a non-controlled background.

Fig. 7 showed testing accuracy on volunteers who use sign language in their daily life are higher compared to volunteers who are non-sign language users. This indicated that the model used is unable to detect signs that are not demonstrated in a standard manner, for example using left hand instead of right hand. Since non-signers are not used to using sign language, they were instructed to follow the tutorial in [27] which was suggested by MDF. This can be the reason for the variation in prediction.

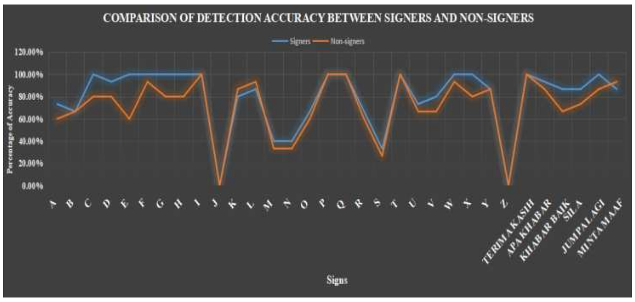


Fig. 7. Graph of comparison of detection accuracy between signers and non-signers

For alphabet A-Z, the average accuracy for signers and non-signers are 76.41% and 68.72% respectively. For action signs, the average accuracy is approximately 92.22% for signers and 84.45% for non-signers. Table I showed samples of action signs by signers and non-signers while Table II showed the average recognition accuracy for the 6 action signs between signers and non-signers.

TABLE I. SAMPLE FROM THE TESTING PHASE

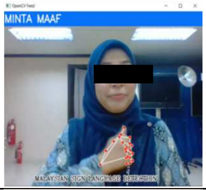
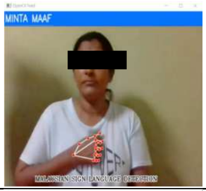
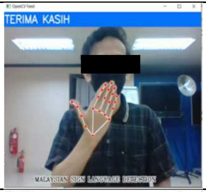
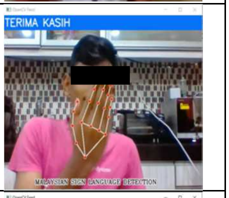
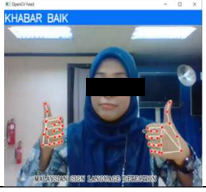
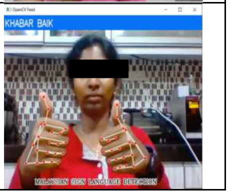
SIGN	SIGN LANGUAGE USERS	NON-SIGN LANGUAGE USERS
MINTA MAAF	 [redacted] OpenCV Feed MINTA MAAF MALAYSIAN SIGN LANGUAGE DETECTION	 [redacted] OpenCV Feed MINTA MAAF MALAYSIAN SIGN LANGUAGE DETECTION
TERIMA KASIH	 [redacted] OpenCV Feed TERIMA KASIH MALAYSIAN SIGN LANGUAGE DETECTION	 [redacted] OpenCV Feed TERIMA KASIH MALAYSIAN SIGN LANGUAGE DETECTION
KHABAR BAIK	 [redacted] OpenCV Feed KHABAR BAIK MALAYSIAN SIGN LANGUAGE DETECTION	 [redacted] OpenCV Feed KHABAR BAIK MALAYSIAN SIGN LANGUAGE DETECTION

TABLE II. PERCENTAGE OF ACCURACY

Signs	Sign Language Users		Non-Sign Language Users	
	Number of successful recognitions over 15 attempts	Accuracy	Number of successful recognitions over 15 attempts	Accuracy
TERIMA KASIH	15	100.00%	15	100.00%
APA KHABAR	14	93.33%	13	86.67%
KHABAR BAIK	13	86.67%	10	66.67%
SILA	13	86.67%	11	73.33%
JUMPA LAGI	15	100.00%	13	86.67%
MINTA MAAF	13	86.67%	14	93.33%

Other alphabets that the volunteers found hard were M and N. This is because of the similarity in the signs (refer to Fig. 7). Fig. 8 showed example of false recognition where alphabet N is recognised as alphabet M. The model failed to recognition alphabet J and Z as both signs require continuous movement and the model only captures and process one frame at a time.

Despite having a small amount of training data, the accuracies achieved are acceptable. Beside testing recognition performance, all volunteers were asked to fill a survey. According to the ISO 9241-11 standard, user satisfaction, together with effectiveness and efficiency, should contribute to usability [28]. 70% of volunteers had never heard of a sign language recognition system which shows that there is a huge gap in the usage of product of new technologies in daily life than the advancement of today's technologies. Half of the volunteers are satisfied with the simplicity of use of the recognition system.

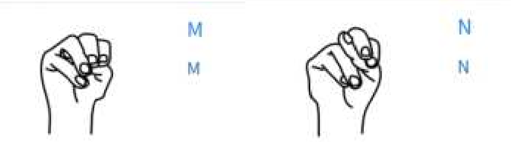


Fig. 8. Sign of the alphabet M and N [6]

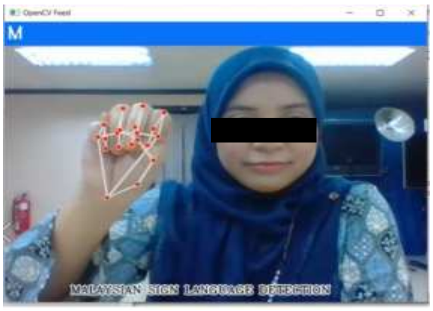

[redacted]

Fig. 9. Incorrect recognition as M instead of alphabet N.

V. CONCLUSION

This paper reported a Malaysian Sign Language Recognition system based on deep learning techniques which uses 3D hand point estimation and LSTM. Experiments recognition accuracy of 84.45% for action signs by non-signers shows promising result to support the continuous work to create a real-time Malaysian Sign Language recognition which will bring non-signers closer to sign language users. The use of new technology is hoped to alleviate the scarcity of BIM interpreters for the benefit of hearing-impaired persons.

ACKNOWLEDGMENT

The authors wish to thank the Faculty of Computer Science and Information Technology, Universiti Malaysia Sarawak for the support.

REFERENCES

[1] Choong, V. Y. (2018). Development of Malaysian Sign Language in Malaysia. *Journal of Special Needs Education*, 8, 15-24. Retrieved from https://journal.nase.org.my/index.php/jsne/article/view/11

[2] Ow, S. H., Mokhtar, S., & Zainuddin, R. (2007). A Review on the teaching and learning resources for the deaf community in Malaysia. *Chiang Mai University Journal of Natural Science*, 1(1), 165-176. Retrieved from https://www.thaiscience.info/Journals/Article/CMUS/10325206.pdf

[3] Hurlbut, H. M. (2003). Cross-linguistic perspectives in sign language research: selected papers from TISLR 2000. In A. Baker, B. van den Bogaerde and O.A. Crasborn (Eds.), *A preliminary survey of the signed languages of Malaysia* (pp. 31-46). Gallaudet University Press.

[4] Awaludin, F. (2021, March 1). *Signing the deaf and mute away from the margins*. MalaysiaNow. https://www.malaysianow.com/news/2021/03/17/signing-the-deaf-and-mute-away-from-the-margins/

[5] Mittal, A., Kumar, P., Roy, P. P., Balasubramanian, R., & Chaudhuri, B. B. (2019). A modified LSTM model for continuous sign language recognition using leap motion. *IEEE Sensors Journal*, 19(16), 7056-7063. doi: 10.1109/JSEN.2019.2909837.

[6] Asri, M. A. M. M., Ahmad, Z., Mohtar, I. A., & Ibrahim, S. (2019). A Real Time Malaysian Sign Language Detection Algorithm Based on YOLOv3. *International Journal of Recent Technology and Engineering*, 8(2), 651-656.

[7] Lien, A. L. S., & Yin, L. K (2020). Gesture Recognition-Malaysian Sign Language Recognition using Convolutional Neural Network. *International Conference on Digital Transformation and Applications (ICDXA)*, 1-6. https://www.tarc.edu.my/files/icdxa/FAA3D03D-6F56-4734-BE7B-0C604EF96422.pdf

[8] Borg, M., & Camilleri, K. P. (2019). Sign language detection "in the wild" with recurrent neural networks. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1637-1641. doi: 10.1109/ICASSP.2019.8683257.

[9] Vo, D. H., Huynh, H. H., Doan, P. M., & Meunier, J. (2017). Dynamic gesture classification for Vietnamese sign language recognition. *International Journal of Advanced Computer Science and Applications*, 8(3), 412-420. http://dx.doi.org/10.14569/IJACSA.2017.080357

[10] Swee, T. T., Salleh, S. H., Ariff, A. K., Ting, C. M., Seng, S. K., & Huat, L. S. (2007). Malay sign language gesture recognition system. *International Conference on Intelligent and Advanced Systems*, pp. 982-985. doi: 10.1109/ICIAS.2007.4658532.

[11] Almasre, M. A., & Al-Nuaim, H. (2020). A comparison of Arabic sign language dynamic gesture recognition models. *Heliyon*, 6(3). https://doi.org/10.1016/j.heliyon.2020.e03554

[12] Alrubayi, A. H., Ahmed, M. A., Zaidan, A. A., Albahri, A. S., Zaidan, B. B., Albahri, O. S., ... & Alazab, M. (2021). A pattern recognition model for static gestures in Malaysian sign language based on machine learning techniques. *Computers & Electrical Engineering*, 95. https://doi.org/10.1016/j.compeleceng.2021.107383

[13] Che, Y., Song, Y., & Qi, Y. (2019, May). A novel framework of hand localization and hand pose estimation. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2222-2226. doi: 10.1109/ICASSP.2019.8682382.

[14] Jang, Y., Noh, S. T., Chang, H. J., Kim, T. K., & Woo, W. (2015). 3D Finger Cape: Clicking action and position estimation under self-occlusions in egocentric viewpoint. *IEEE Transactions on Visualization and Computer Graphics*, 21(4), 501-510. doi: 10.1109/TVCG.2015.2391860.

[15] Chang, H. J., Garcia-Hernando, G., Tang, D., & Kim, T. K. (2016). Spatio-temporal hough forest for efficient detection–localisation–recognition of fingerwriting in egocentric camera. *Computer Vision and Image Understanding*, 148, 87-96. https://doi.org/10.1016/j.cviu.2016.01.010

[16] Yang, J., Chang, H. J., Lee, S., & Kwak, N. (2020, August). SeqHAND: RGB-sequence-based 3D hand pose and shape estimation. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, JM. (eds). Computer Vision – ECCV 2020. ECCV 2020. *Lecture Notes in Computer Science*, vol. 12357. Springer, Cham. https://doi.org/10.1007/978-3-030-58610-2_8

[17] Huang, L., Zhang, B., Guo, Z., Xiao, Y., Cao, Z., & yuan, J. (2021). Survey on depth and RGB image-based 3D hand shape and pose estimation. Virtual Reality & Intelligent Hardware, 3(3), 207-234. https://doi.org/10.1016/j.vrih.2021.05.002

[18] Oikonomidis, I., Kyriazis, N., & Argyros, A. A. (2011). Efficient model-based 3D tracking of hand articulations using Kinect. In Jesse Hoey, Stephen McKenna and Emanuele Trucco, *Proceedings of the British Machine Vision Conference*, pp. 101.1-101.11. BMVA Press. http://dx.doi.org/10.5244/C.25.101

[19] Qian, C., Sun, X., Wei, Y., Tang, X., & Sun, J. (2014). Realtime and robust hand tracking from depth. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1106-1113. doi: 10.1109/CVPR.2014.145.

[20] Zimmermann, C., & Brox, T. (2017). Learning to estimate 3D hand pose from single RGB images. *IEEE International Conference on Computer Vision*, pp. 4903-4911. doi: 10.1109/ICCV.2017.525.

[21] Sharma, S., & Huang, S. (2021). An end-to-end framework for unconstrained monocular 3D hand pose estimation. *Pattern Recognition*, 115. https://doi.org/10.1016/j.patcog.2021.107892

[22] Elboushaki, A., Hannane, R., Afdel, K., & Koutti, L. (2020). Improving articulated hand pose detection for static finger sign recognition in RGB-D images. *Multimedia Tools and Applications*, 79(39), pp. 28925-28969. https://doi.org/10.1007/s11042-020-09370-y

[23] Sridhar, S., Mueller, F., Oulasvirta, A., & Theobalt, C. (2015). Fast and robust hand tracking using detection-guided optimization. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3213-3221. doi: 10.1109/CVPR.2015.7298941.

[24] Github. MediaPipe v0.7 – Hands (2020). https://google.github.io/mediapipe/solutions/hands.html

[25] Eckle, K., & Schmidt-Hieber, J. (2019). A comparison of deep networks with ReLU activation function and linear spline-type methods. *Neural Networks*, 110, 232-242. https://doi.org/10.1016/j.neunet.2018.11.005

[26] Jiang, X., Hu, B., Chandra Satapathy, S., Wang, S. H., & Zhang, Y. D. (2020). Fingerspelling identification for Chinese sign language via AlexNet-based transfer learning and Adam optimizer. *Scientific Programming*, vol. 2020. https://doi.org/10.1155/2020/3291426

[27] Persekutuan Orang Pekak Malaysia (2021). Bahasa Isyarat Malaysia. https://www.bimsignbank.org/

[28] Lindgaard, G., & Dudek, C. (2002). User satisfaction, aesthetics and usability. In: Hammond, J., Gross, T., Wesson, J. (eds) Usability. IFIP WCC TC13 2002. *IFIP — The International Federation for Information Processing*, vol 99. Springer, Boston, MA. https://doi.org/10.1007/978-0-387-35610-5_16