Arabic Sign Language Letter Recognition using MobileNet-v2 Deep Neural Network

[1]Mohamed Hamdy Abdallah Eissa, [2]Alaa Ismaeel, [3]Sherimon P.C., [4]Vinu Sherimon, [5]Remya Revi K.

[1,2,3]Faculty of Computer Studies, [4]Information Technology Department, [5] Computer Science and Engineering Department
[1,2,3]Arab Open University, Muscat, Sultanate of Oman
[4]University of Technology and Applied Sciences, Muscat, Sultanate of Oman
[5]Saintgits College of Engineering, Kottayam, India

Abstract : An intelligent Arabic Sign Language Recognition system is developed using the deep learning technique. A pre-trained lightweight deep Convolutional Neural Network architecture called MobileNet-v2 is used in this work. The network is fine-tuned using the transfer learning approach to classify thirty-two different Arabic sign language letters. Recently published benchmarked Arabic Sign Language Letter (ArSL21L) is used for the experimental analysis. The model achieved a good classification performance with an F1-score of 0.808.

Index Terms : Sign Language Recognition, Deep Learning, Arabic Sign Language Letters, Transfer Learning, Fine-tuning

I. Introduction

The people with disabilities in hearing and speaking face difficulties when communicating with other people since the sign language is not easily understood by everybody. This problems in communication can be overcome by developing an efficient Sign Language Recognition (SLR) technique. In this research work, we focus on developing an intelligent Arabic Sign Language (ASL) letter recognition system. This work is an initial study of a project to develop a smart glove which interprets the ASL into text. As an initial work, we investigate the usage of deep learning technique in machine learning to develop an intelligent recognition system to recognize the ASL letters. Hence, in this work, we design and implement a multi-class deep-learning based classification model for Arabic Sign Language Recognition (ASLR). A lightweight pre-trained deep convolutional neural network, MobileNet-v2 [1] is fined tuned using transfer learning approach. The proposed method is evaluated using the recently published benchmarked and challenging ASLR dataset called Arabic Sign Language Letter (ArSL21L) dataset [2].

The upcoming sections discuss about the related works, proposed method, experimental result analysis, and conclusion.

II. Related Works

The initial activity in this research would be to identify and study similar research undertaken on helping aids for Deaf / Mute Individuals. A brief review of the research works related to SLR and ASLR are given below.

Zaki et al. used a Hidden Markov Model to recognize the ASL letters [3]. Shingate et al. proposed a smart system that can translates “voice into text and text to sign language” automatically [4]. Halder and Tayade used a Convolutional Neural Network (CNN) for real time translation of sign language [5]. Rwelli et al. designed a gesture based ASLR system which uses a deep convolutional neural network [6]. Ismail et al. created an ASLR database using Kinect V2 sensor, and used a deep-learning based multi-modal fusion technique to recognize the ASL letters [7].

Boukdir et al. proposed a method based on a deep learning technique to recognize Moroccan Arabic sign language letters from videos. They design two approaches using on a Convolutional Recursive Neural Network and 3D CNN [8]. Al-Khuraym and Ismail used a lightweight CNN model called EfficientNet-Lite 0 to develop a ASLR system [9]. Alsaadi et al. proposed a real time method based on deep learning technique using AlexNet architecture to recognize Arabic sign language letters [10]. Duwairi and Halloush [11] proposed an automatic ASLR model using VGGNet architecture and the model is evaluated using Arabic alphabets Sign Language (ArSL2018) dataset [12].

III. Proposed Method

The ASLR is a multiclass classification problem, with thirty-two classes. Here, a deep-learning based classification model is developed for this multiclass classification problem, and the outline of the proposed method is shown in Fig. 1.

The deep learning techniques using CNNs give remarkable results in the computer vision tasks [13-15]. A CNN is designed to mimic the human visual system and basically used in image recognition tasks [16]. To obtain accurate and reliable classification results, a CNN should be trained on huge quantity of labelled image data. But, in most real-life circumstances, there is a shortage in

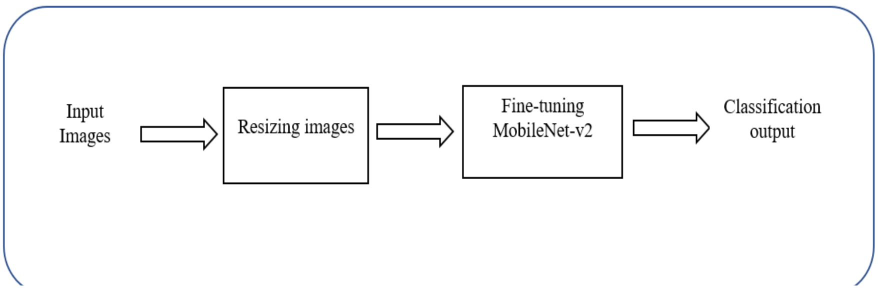

Fig. 1 The overview of the proposed method

the availability of the number of labelled data, and hence, training the network from scratch is a difficult procedure. So, to overcome this problem, we can utilize a pre-trained CNN using the transfer learning approach [17-19]. The ASLR is also a small sample size classification problem, and hence, we use the transfer learning approach to develop the classification model. In this work, we utilize a lightweight pre-trained CNN architecture called MobileNet-v2. This network is pre-trained on a huge dataset called ImageNet dataset. The specifications of the network are: 3.5 million learnable parameters, depth is 53, size is 13 MB, input size is 224 x 224 x3 [1]. The layers of this pre-trained network learned the various basic features as well as discriminative features [20]. Hence, we fine-tune the network to classify the thirty-two categories in the ASLR problem. To fine-tune the network using the transfer learning, as a first step we remove the Fully Connected (FC) layer of the MobileNet-v2, and is replaced with a new FC layer for ASLR problem. Then, we train the reorganized network using the ArSL21L dataset for the thirty-two-class classification.

IV. RESULTS AND DISCUSSION

The experiments are carried out on GPU based system with NVIDIA GTX 1060 card, 6.0 GB RAM using MATLAB platform with help of Deep Learning Toolbox. The next subsections briefs about the dataset used, evaluation metrics, training options, and experiments results and analysis.

Dataset Used

The benchmarked publicly available dataset, ArSL21L dataset [2] is used for the Arabic sign language classification. The dataset contains thirty-two Arabic sign language letters as shown in Fig. 2. This recently published ArSL21L dataset contains train dataset with 9,955 images and test dataset with 4,247 images. The resolution of the images is 416 x 416 x 3, and the images are in 'jpg' format.

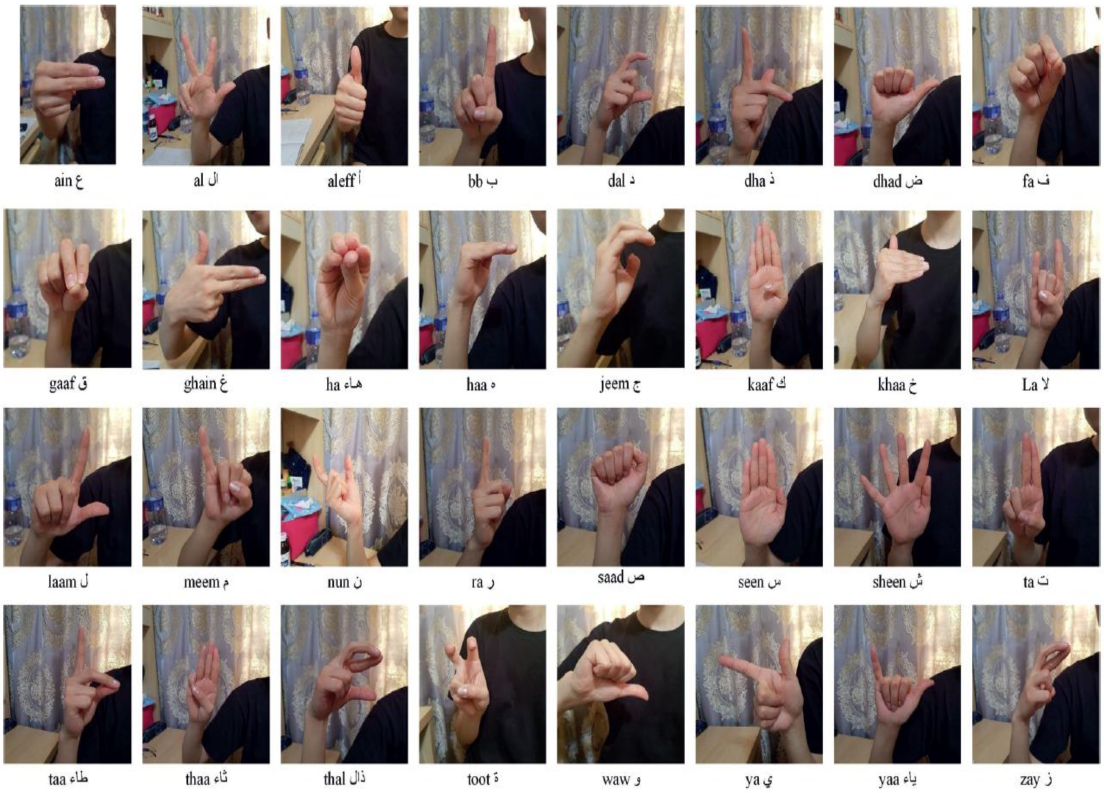

Fig. 2 The thirty-two Arabic sign language letters from the ArSL21L dataset [2]

Evaluation Policies

Precision, Recall, and F1-score are three evaluation metrics used for assessing the performance of the model, and are calculated based on the Equations (1), (2), and (3), respectively. The values of True Negative (TN), True Positive (TP), False Positive (FP), and False Negative (FN) are obtained from the confusion matrix.

$$Precision = \frac{TP}{(TP+FP)} \quad (1)$$

$$Recall = \frac{TP}{(TP+FN)} \quad (2)$$

$$F1 - score = \frac{2\times(Precision\times Recall)}{(Precision+Recall)} \quad (3)$$

Training Options

The options used to train the network is provided in Table 1. From the training dataset, 70 % are used for training the model and 30% are used for validating the model. The size of the images is reduced to 224 x 224 x 3 resolution based on the input size of the MobileNet-v2 network.

Table 1 Training options

Learning rate	0.0001
No. of epochs	60
Mini-batch size	64

Optimizer	adam

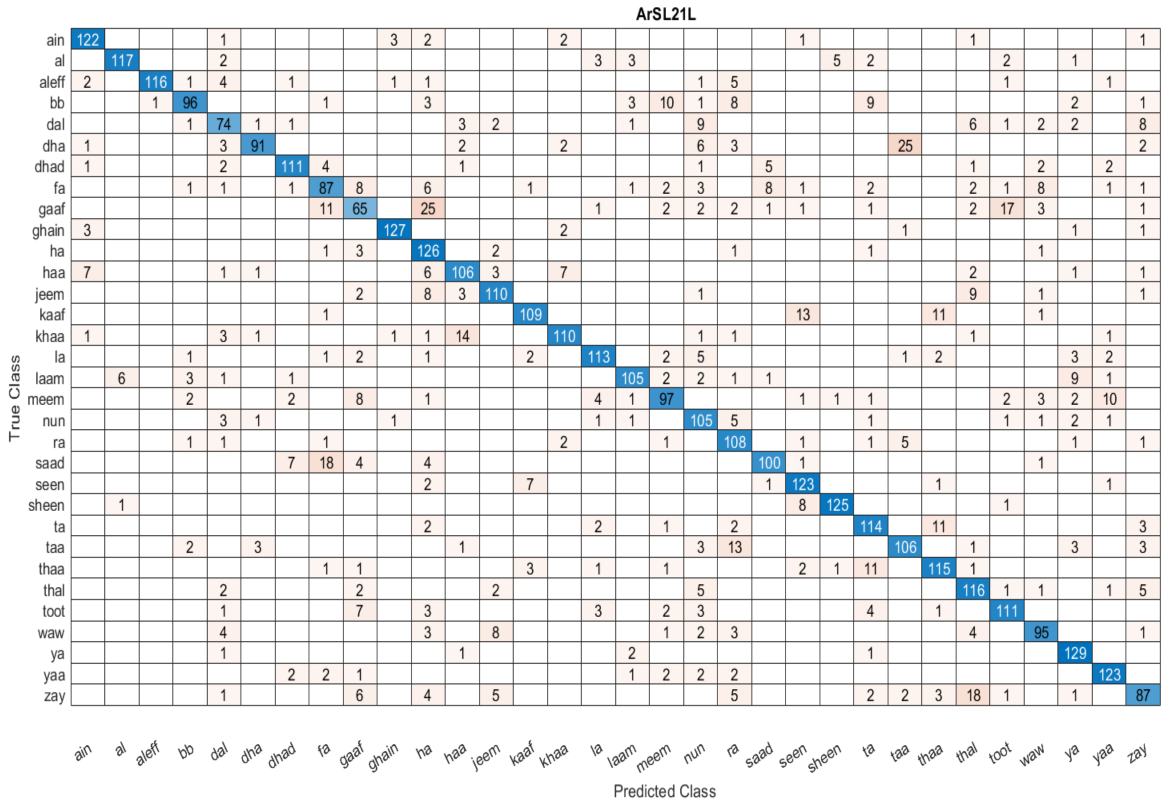

Fig. 3 The confusion matrix of the model

V. EXPERIMENTAL RESULTS AND ANALYSIS

The MobileNet-v2 network is fine-tuned on ArSL21L dataset for classifying the Arabic sign language letters into the thirty-two categories using the above-mentioned training options. The performance of the model is tested using the images from the test dataset. Fig. 3 shows the confusion matrix of the model. Then, the values of Precision, Recall, and F1-score values of each class are evaluated from the confusion matrix, and is given in Table 2. The analysis from the table shows that, the average values of Precision, Recall, and F1-score achieved by the model are 0.809, 0.815, and 0.808, respectively. This shows the good classification performance of the model. The class ‘ghain’ achieves the highest performance with an F1-score of 0.948, but, the class ‘gaaf’ achieves a comparatively low F1-score of 0.535. This shows that even though the model achieves a good performance, still there is a need of further improvement.

Table 2 Performance evaluation of the model

SL No.	Class Name	Precision	Recall	F1-score
1	“ain”	0.917	0.891	0.904
2	“al"	0.867	0.944	0.904
3	“aleff”	0.866	0.991	0.924
4	“bb”	0.711	0.889	0.790
5	“dal”	0.667	0.705	0.685
6	“dha”	0.674	0.929	0.781
7	“dhad”	0.854	0.881	0.867
8	“fa”	0.644	0.680	0.662
9	“gaaf”	**0.485**	**0.596**	**0.535**
10	“ghain”	**0.941**	**0.955**	**0.948**
11	“ha”	0.933	0.636	0.756
12	“haa”	0.785	0.809	0.797
13	“jeem”	0.815	0.833	0.824
14	“kaaf”	0.807	0.893	0.848
15	“khaa”	0.815	0.880	0.846
16	“la”	0.837	0.883	0.859
17	“laam”	0.795	0.890	0.840
18	“meem”	0.719	0.789	0.752
19	“nun”	0.854	0.691	0.764
20	“ra”	0.878	0.679	0.766
21	“saad”	0.749	0.862	0.802

22	"seen"	0.911	0.809	0.857
23	"sheen"	0.926	0.947	0.936
24	"ta"	0.844	0.760	0.800
25	"taa"	0.785	0.757	0.771
26	"thaa"	0.839	0.799	0.819
27	"thal"	0.859	0.709	0.777
28	"toot"	0.822	0.799	0.810
29	"waw"	0.785	0.798	0.791
30	"ya"	0.963	0.822	0.887
31	"yaa"	0.911	0.854	0.882
32	"zay"	0.644	0.744	0.690
	Average	**0.809**	**0.815**	**0.808**

VI. CONCLUSION

In this work, we investigated the use of deep learning technique in machine learning to recognize the ASL letters. A pre-trained CNN architecture called MobileNet-v2 is fine-tuned using the transfer learning approach to classify the thirty-two ASL letters. The model is evaluated using the recently published benchmarked ArSL21L dataset. The model achieved a good classification performance with an F1-score of 0.808.

VII. ACKNOWLEDGMENT

The research leading to these results has received funding from the Research Council (TRC) of the Sultanate of Oman under the Block Funding Program "BFP/URG/ICT/21/155".

REFERENCES

1. A. G. Howard et al., "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications." pp. 1–9, 2017. [Online]. Available: http://arxiv.org/abs/1704.04861
2. G. Batnasan, M. Gochoo, M. E. Otgonbold, F. Alnajjar, and T. K. Shih, "ArSL21L: Arabic Sign Language Letter Dataset Benchmarking and an Educational Avatar for Metaverse Applications," in IEEE Global Engineering Education Conference, EDUCON, 2022, pp. 1814–1821.
3. M. Zaki, A. Mahmoud, S. Abd, and E. Mostafa, "Arabic Alphabet and Numbers Sign Language Recognition," Int. J. Adv. Comput. Sci. Appl., vol. 6, no. 11, pp. 209–214, 2015, doi: 10.14569/ijacsa.2015.061127.
4. D. S. Shingate, R. Bajaj, A. Singh, W. Gayatri, and Y. Bhavar, "Sign Language Interpretation," Int. J. Sci. Dev. Res., vol. 4, no. 10, pp. 89–92, 2019.
5. A. Halder and A. Tayade, "Sign Language to Text and Speech Translation in Real Time Using Convolutional Neural Network," Int. J. Eng. Res. Technol., vol. 8, no. 15, pp. 191–196, 2020.
6. R. El Rwelli, O. R. Shahin, and A. I. Taloba, "Gesture based Arabic Sign Language Recognition for Impaired People based on Convolution Neural Network," Int. J. Adv. Comput. Sci. Appl., vol. 12, no. 12, pp. 574–582, 2021, doi: 10.14569/IJACSA.2021.0121273.
7. M. H. Ismail, S. A. Dawwd, and F. H. Ali, "Dynamic hand gesture recognition of Arabic sign language by using deep convolutional neural networks," Indones. J. Electr. Eng. Comput. Sci., vol. 25, no. 2, pp. 952–962, 2022, doi: 10.11591/ijeecs.v25.i2.pp952-962.
8. A. Boukdir, M. Benaddy, A. Ellahyani, O. El Meslouhi, and M. Kardouchi, "Isolated Video-Based Arabic Sign Language Recognition Using Convolutional and Recursive Neural Networks," Arab. J. Sci. Eng., vol. 47, no. 2, pp. 2187–2199, 2022, doi: 10.1007/s13369-021-06167-5.
9. B. Y. Al-Khuraym and M. M. Ben Ismail, "Arabic Sign Language Recognition using Lightweight CNN-based Architecture," Int. J. Adv. Comput. Sci. Appl., vol. 13, no. 4, pp. 319–328, 2022, doi: 10.14569/IJACSA.2022.0130438.
10. Z. Alsaadi, E. Alshamani, M. Alrehaili, A. A. D. Alrashdi, S. Albelwi, and A. O. Elfaki, "A Real Time Arabic Sign Language Alphabets (ArSLA) Recognition Model Using Deep Learning Architecture," Computers, vol. 11, no. 5, pp. 1–20, 2022, doi: 10.3390/computers11050078.
11. R. M. Duwairi and Z. A. Halloush, "Automatic recognition of Arabic alphabets sign language using deep learning," Int. J. Electr. Comput. Eng., vol. 12, no. 3, pp. 2996–3004, 2022, doi: 10.11591/ijece.v12i3.pp2996-3004.
12. G. Latif, N. Mohammad, J. Alghazo, R. AlKhalaf, and R. AlKhalaf, "ArASL: Arabic Alphabets Sign Language Dataset," Data Br., vol. 23, pp. 1–4, 2019, doi: 10.1016/j.dib.2019.103777.
13. N. Sharma, V. Jain, and A. Mishra, "An Analysis of Convolutional Neural Networks for Image Classification," in Procedia Computer Science, 2018, vol. 132, pp. 377–384. doi: 10.1016/j.procs.2018.05.198.
14. F. Sultana, A. Sufian, and P. Dutta, "Advancements in image classification using convolutional neural network," in 2018 Fourth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), 2018, pp. 122–129. doi: 10.1109/ICRCICN.2018.8718718.
15. L. Chen, S. Li, Q. Bai, J. Yang, S. Jiang, and Y. Miao, "Review of image classification algorithms based on convolutional neural networks," Remote Sens., vol. 13, no. 22, pp. 1–51, 2021, doi: 10.3390/rs13224712.
16. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," Commun. ACM |, vol. 60, no. 6, pp. 84–90, 2017.
17. A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "CNN features off-the-shelf: An astounding baseline for recognition," in IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2014, pp. 512–519. doi: 10.1109/CVPRW.2014.131.

18. J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, “How transferable are features in deep neural networks?,” in Advances in Neural Information Processing Systems, 2014, pp. 3320–3328.
19. N. Becherer, J. Pecarina, S. Nykl, and K. Hopkinson, “Improving optimization of convolutional neural networks through parameter fine-tuning,” Neural Comput. Appl., vol. 31, no. 8, pp. 3469–3479, 2019, doi: 10.1007/s00521-017-3285-0.
20. M. Oquab, L. Bottou, I. Laptev, and J. Sivic, “Learning and transferring mid-level image representations using convolutional neural networks,” in IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2014, pp. 1717–1724. doi: 10.1109/CVPR.2014.222.