Arabic Sign Language Recognition through Deep Neural Networks Fine-Tuning

https://doi.org/10.3991/ijoe.v16i05.13087

Yaser Saleh (✉), Ghassan F. Issa
University of Petra, Amman, Jordan
Yaser.Saleh@uop.edu.jo

Abstract—Sign Language is considered the main communication tool for deaf or hearing-impaired people. It is a visual language that uses hands and other parts of the body to provide people who are in need to full access of communication with the world. Accordingly, the automation of sign language recognition has become one of the important applications in the areas of Artificial Intelligence and Machine learning. Specifically speaking, Arabic sign language recognition has been studied and applied using various intelligent and traditional approaches, but with few attempts to improve the process using deep learning networks. This paper utilizes transfer learning and fine tuning deep convolutional neural networks (CNN) to improve the accuracy of recognizing 32 hand gestures from the Arabic sign language. The proposed methodology works by creating models matching the VGG16 and the ResNet152 structures, then, the pre-trained model weights are loaded into the layers of each network, and finally, our own soft-max classification layer is added as the final layer after the last fully connected layer. The networks were fed with normal 2D images of the different Arabic Sign Language data, and was able to provide accuracy of nearly 99%.

Keywords—Arabic Sign Language Recognition, Deep Learning, Fine tuning, Convolutional neural network.

1 Introduction

According to the World Health Organization in 2019, the number of people with hearing disability is around 466 million with 34 million of these are children [1]. It is also estimated that this number is expected to double in the next 30 years. These numbers show the importance of sign language as a tool for communication between hearing impaired people and the rest of the world. Sign Language has been improved and standardized by many different countries and different cultures resulting in different standards such as the American Sign Language (ASL), British Sign Language (BSL), Arabic Sign Language, and others. Unfortunately, there is no universal sign language. As this paper focuses on Arabic sign language, we found that there is no one standard Arabic sign language, but instead many variations as many as the Arabic-speaking countries. Luckily enough, these different Arabic sign languages share the same signs for alphabets [2].

It is worth mentioning here, that research on sign language recognition requires different areas of expertise including sign extraction, sign feature representation, sign classification, and much more [3]. This is in addition to the fact that there is no single universal sign language [4].

The motivation of the work presented in this paper is to engage in the efforts to find viable solutions for the automation of sign language recognition, thus reducing the need for human intervention during the interpretation. The core of the work presented here is to be able to recognize sign language through feeding a convolutional neural network (CNN) with images of hand gestures in different lighting conditions and different orientations. The use of deep neural networks has recently boosted many research areas in image classification, such as medical images classification, objects recognition, face and ID recognition, and much more [5, 6]. Looking at the ImageNet challenge results in the first five years of the competition; one can easily see the progress in the reduction of classification error rates using different models of deep neural networks [7].

With the recent developments in Convolutional Neural Networks, improvements on the overall capabilities of multiple image processing areas displayed great results. One of the top leading network models for the ImageNet challenge was the Inception-v4 model architecture, which was able to achieve 3.08% top error on the ImageNet challenge through using 75 trainable layers [8], overcoming the ResNet and GoogleNet that were the previous champions in image classification [9]. However, one of the main downsides of latest state of the art is the increasing training run time, and the very large datasets required for training and testing.

The proposed method in this paper is created to be suitable to operate on image data of Arabic sign language gestures captured in different lighting conditions and different backgrounds [10]. The objective of this paper is to utilize a dataset of 32 different Arabic signs images, and to present how fine tuning can help in training a network with a batch of images, and provide higher accuracy compared to other existing techniques.

This paper is divided as follows: Section 1 contains the introduction, Section 2 comments on some related work, Section 3 explains the CNN used, and the procedure of the experiments, Section 4 discusses the results, and finally Section 5 presents the conclusion.

2 Related Work

Nowadays, the modern society is a witness to the power of machine learning technology that ranges from web searches to recommendations on e-commerce websites. The machine learning systems that are developed everyday provide help in object identification of images, speech transcription, and even in the selection of results of a search [11]. The use of deep learning has made it possible to transfer the benefits of the machine learning technology to provide better solutions for the Arabic sign language problem.

One of the key features in sign language has been hand gestures recognition, where research provided different ideas regarding reading the input of the gestures, starting with the use of digital imaging and digital cameras. Many researchers have worked on

the area of gesture recognition using more traditional approaches that did not employ machine or deep learning methods. For example, in 2013, Singha and Das presented an approach for identifying different alphabets of Indian Sign Language, through a proposed system that comprised of three stages, first, a preprocessing stage, then feature extraction and finally classification. The preprocessing stage consisted of skin filtering and histogram matching. The feature extraction was implemented through Eigen values and Eigen Vectors, and finally Eigen value weighted Euclidean distance was used for classification. 24 different alphabets were considered and a 96.25% success rate was obtained [12]. Other researchers such as Nachmai used a special algorithm called SIFT (Scale Invariant Feature Transform) aimed at recognizing English alphabets [13]. The novelty of their approach is that, it is a space, size, illumination, and rotation invariant. Other approaches to sign language recognition are based on the Hidden Markov Models such as the work done by Youssif in 2011 which recognizes Arabic Sign Language with an accuracy reaching up to 82.22% [14]. Another work that employed the Hidden Markov Models can be seen in [15], while in [16] a five stages process was presented for an Arabic sign language translator, focusing on feature extraction of translation, scale, and rotation invariant, and presenting accuracy of 91.3%.

The use of special sensors such as the Microsoft Kinect and Leap Motion Sensors were used by Almasre and Al-Nuaim for data acquisition in their hand-gesturing model with the purpose of recognizing 28 Arabic Sign Language gestures [17]. Chuan and Guardino, on the other hand, used only a Leap Motion Sensor with k-nearest neighbor and support vector machine algorithms in order to classify the 26 letters of the English alphabet in American Sign Language [18]. ElBadawy in 2015 also used a Leap Motion Sensor in combination with two digital cameras to capture the data, where 20 signs for different words were inputted into a feature extraction and sign language translation algorithm for classification and reproduction of text data [19].

The recent advancements in machine learning and deep learning, and the outstanding results obtained from Convolutional Neural Networks (CNN) in image classification and prediction, paved the way for researchers to apply them in hundreds of applications including sign language recognition. For example, Kang in 2015 used data provided by the Microsoft Kinect with CNN and presented a methodology for recognizing fingerspelling, and tested their technique on the American Sign Language [20]. The data consisted of 31 alphabets and numbers, where the proposed system was inputted the depth maps of the Kinect and was able to reach a maximum accuracy of 85.5%. Another example for the use of Kinect depth maps and CNNs was shown in [21], where the technique was applied on Italian Sign Language dataset consisting of 20 gestures. The technique presented two main models, one for upper-body features extraction and another for hand features extraction, where results were merged and inputted into a neural network classifier. The proposed technique achieved an accuracy of 91.7%.

Priyadharsini and Rajeswari, in 2017 used CNNs with a dataset of 26 Indian alphabet images for sign language recognition, achieving an accuracy that reached 100% [22]. Convolutional Neural Networks were also used with a new model for recognizing 10 Korean words taken from video frames, the model was created in three different stages and was capable of acquiring accuracy up to 84.5% [23].

Recent work relating to Arabic sign language recognition was shown in [24], where different CNNs were built and fed with data from a depth sensor which included not only height and width, but also the depth of objects. The data then is passed through a CNN depending on the frame rate of the depth video, which also controls how deep the network is. For lower frame rates, lower depth is used, while higher frame rate means more depth. The proposed approach acquired accuracy of up to 98%.

Hayani in 2019 presented a new model for Arabic Sign Language Recognition through the use of Convolutional Neural Networks for recognizing 28 Arabic letters and numbers from 0 to 10 using an image dataset containing a total of 7869 images. The proposed model contained 7 layers, and was trained multiple times on different training-testing variations, with highest accuracy being 90.02% with a training set size of 80% of images, finally the authors presented a comparison with other techniques showing the proposed model advantage [25].

A combination of Convolutional Neural Networks and Long-Term Memory network was presented in [26], the goal was to recognize Chinese sign language gestures collected as frames from videos, and the proposed technique provided an accuracy of 95% for recognizing 40 words.

While Convolutional Neural Networks seems to work very well with image recognition, one of its drawbacks is the large amounts of data required to train the network, thus requiring excessive amount of time and processing capabilities. In order to reduce the size of the data set, and processing time, researchers employed Fine-Tuning techniques and Transfer Learning, relying on previously trained networks with large scale datasets (general concepts), and fine tune the network with a small scale and more specific dataset in order to achieve high accuracy rates.

A comparison between the use and non-use of fine tuning in image recognition was presented by [6]. The AlexNet model was implemented as an example of a well-known model that can be used with or without fine tuning. The authors focused on medical images to present a challenge as normally pre-trained networks are trained on natural images. The paper provided evidence showing how fine tuning can help in maximizing accuracy without the need for large datasets and training from scratch, also, the paper showed how fine tuning several layers could provide different accuracy for different types of images.

A plant recognition model which used fine-tuning with CNNs was presented by [27]. The proposed model was pre-trained on a dataset of 1.2 million images for a set of 1,000 objects classes, then, the model was fine-tuned to recognize 1,000 types of plants, showing how using transfer learning can be a solution to transfer the recognition capabilities of a general domain to a specific one.

Yanai and Kawano, 2015 utilized fine-tuning in a food recognition system [28], with a pre-trained network that was firstly trained on the ImageNet dataset, then fine-tuned using a relatively small dataset of food images. The presented experiment delivered recognition accuracy of 78% and 67% where the authors concluded that more data shall increase accuracy much greatly.

Another example on the use of a CNN and fine-tuning was shown in [29], implementing a CNN loaded with pre-trained weights and training the model with different

datasets of Bangali sign language static images, the datasets included 37 gestures and was able to provide an accuracy of 96.33%.

3 Proposed Procedure

This section presents the neural network models employed including the data collection, under sampling, augmentation, and the fine-tuning process.

3.1 Utilized models

A convolutional neural network (CNN) works by allowing an image to pass through as input, then to go through a set of layers containing convolutions, pooling and other fully connected layers, to finally provide an output of a single class of a set of possible classes for the image. This section of the paper briefly introduces two well-known CNN architectures that have been used with our experiment on the Arabic Sign Language dataset.

One of the already published models with training parameters that has been widely used in recent years is the VGGNet model architecture, which was able to win the ImageNet Challenge in 2014. The VGGNet model architecture was capable of consisting fewer layers than the published state-of-the-art while still providing considerably good results with an error of 7.3% on the overall accuracy [5]. The VGGNet models are also published and widely distributed on the Internet for various deep learning frameworks, making it the preferred model to work with by other researchers. The convolutional layer parameters are denoted as "conv (receptive field size) - (number of channels)" the structure of the used (16 layers) model can be described as seen in Fig. 1.

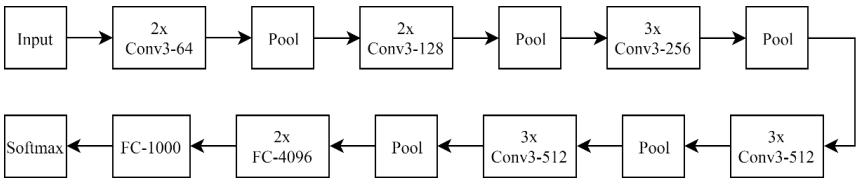

Fig. 1. VGG16 model structure

A year after the introduction of the VGGNet, a new network model called Residual Network (ResNet) was presented by [30]. This model was created with different variations of depth, from 32 layers to 152 showing the increase of accuracy with the increase of depth. The network's structure was combined with a residual function reformulating the layers and providing capabilities for improvement with larger yet less complex networks, the model structure can be seen in Fig. 2. The presented models were able to achieve 3.57% error on the ImageNet test set winning the 1st place in the ImageNet competition in 2015.

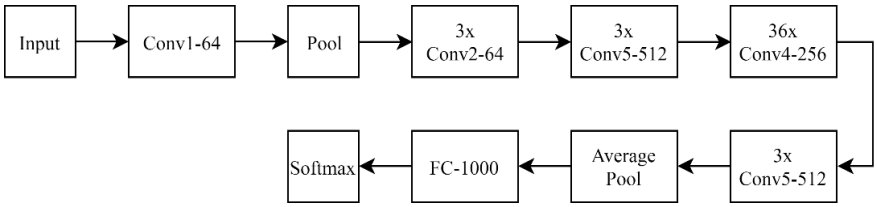

Fig. 2. ResNet152 model structure

3.2 Data preparation

To evaluate the two mentioned models above, a dataset of images (ArASL) presented in [10] was utilized. The dataset originally contained 54,049 images distributed around 32 classes of Arabic Signs, the images dimensions are unified on 64 x 64, and many variations of images were presented through the use of different lighting and backgrounds. A sample of the dataset can be seen in Fig. 3. One issue that must be noted in the above dataset is that the number of images per each class is not the same. In order to solve this problem of imbalance, Under-sampling technique is applied and is described below:

Under-sampling: When a dataset consists of classes that are of different size in terms of data items, a problem of class imbalance occurs resulting in a bias towards the majority class (one having most data items), which negatively affects the performance of the classification. To solve this problem, re-sampling is applied to the data to fix the imbalance and reduce the bias. Under-sampling is one technique of re-sampling that is used to reduce the size of the majority class.

Table 1 which contains some samples of the ArASl dataset, shows for example that one gesture (AIN) consists of 2014 images, while the (YAA) gesture has only 1293 images. Accordingly, Random Under-sampling was applied to the dataset before the use the fine-tuning process. Since the images are not evenly distributed on the classes, random images were discarded from the experiment to even the weight of each class, which also helped in showing the advantage of the use of fine-tuning with smaller datasets. The final dataset included 25,600 images, distributed evenly on 32 classes, giving each class 800 images.

Augmentation: Data Augmentation is a process in which new sets of data are created using existing ones, thus increasing the data diversity for the training process.

As the survey presented by Shorten and Khoshgoftaar have shown, the use of data augmentation can help build a more robust classification network, and can give a boost to the accuracy of the network itself, the authors presented how different types of augmentation can affect the process of training deep convolutional neural networks, starting with how data augmentation provides a reasonable solution for overfitting, which can occur when the training dataset is very small, and how different augmentation techniques can be applied with deep networks to assist the network in achieving better results than when no data augmentation is applied [31].

Random operations are applied to the existing images, and the resulting dataset would then be inputted into the training algorithm with the use of an image augmentation object, this object would apply the following augmentations on the images on every iteration of training:

- Rescaling with a 1/255 factor
- Random horizontal flipping
- Random rotation
- Random height and width shifts
- Random zoom

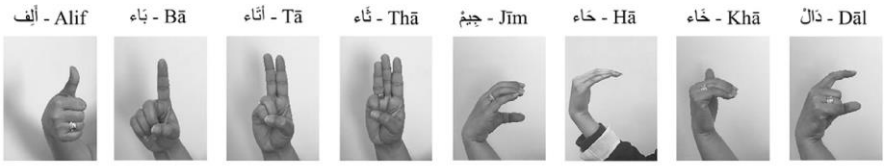

Fig. 3. Arabic Sign Language dataset samples

Table 1. Number of images available in the ArASL dataset for selected gestures

Gesture	Number of images	Gesture	Number of images	Gesture	Number of images
AIN	2014	HA	1592	FA	1955
AL	1343	HAA	1526	GAAF	1705
ALEFF	1671	JEEM	1552	GHAIN	1977
BB	1790	KAAF	1774	NUN	1819
DAL	1634	LA	1746	YAA	1293
DAH	1723	LAAM	1832	ZAY	1374

3.3 Fine-tuning

As mentioned earlier, the VGG16 and the ResNet152 models are chosen for their well-known high performance capabilities, furthermore, the process of fine-tuning the networks will provide the ability of using smaller size dataset such as the one used in this methodology, and will require less number of epochs, representing a pass through the network with the new data.

In our proposed methodology, to build our models for fine tuning we begin by creating models matching the VGG16 and the ResNet152 structures, then, the pre-trained model weights are loaded into the layers of each network, and finally, our own softmax classification layer will be added as the final layer after the last fully connected layer. This process prepared the models for fine-tuning which is executed by running additional training iterations using our own Arabic Sign Language data, an illustration of the steps of fine-tuning can be seen in Fig. 4.

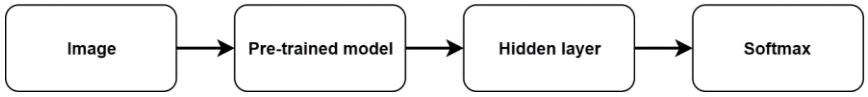

Fig. 4. The fine-tuning process

Following the construction of the models, the Arabic Sign gestures images dataset was used to further train the network, dividing the images such as 80% of the images were used for training and the remaining 20% for testing. The training process included running 100 epochs, passing through the network using the whole dataset, where in each epoch, multiple batches of the image dataset were passed through the network, while correspondingly computing the loss function through categorical cross-entropy between predictions and targets, and running a Stochastic Gradient Descent updates, passing the learning rate of 0.0001, which controlled the size of the update steps, and momentum of 0.9 as parameters. For each epoch, processing runtime, training and validation loss, and the training and validation accuracy are recorded and examined.

4 Experiments, Results, and Analysis

With the established success of the previously mentioned models, the VGG16 and the ResNet152 networks were chosen for the fine-tuning process as described in the preceding section, the dataset used for training consisted of 25,600 images, distributed on 32 classes of Arabic Sign Language gestures with a unified format. The dataset was split into a training set and a validation set with the training set having 80% of the data and 20% was left for the validation set. The full proposed technique would then have the following steps: we begin by under-sampling the data to even out the classes, then the data is inputted into the fine tuning process, where iterations\ epochs are run through the required model (VGG16 \ ResNet152) with data augmentation applied at the beginning of each iteration, and accuracy values being provided at the end of each iteration. Finally, after the final training epoch has been run, the final accuracy is displayed and the training process is finished. The process can be seen in Fig. 5.

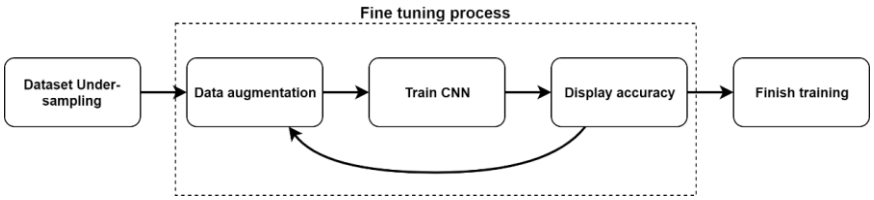

Fig. 5. The proposed methodology

The experimentation started with the VGG16 model, as the data was used in running 100 epoch, passing through the model with our own data and displaying a step-by-step output for each epoch. The model was able to reach a 99% validation accuracy at the 40th epoch, while the highest accuracy of 99.45% was reached at the 92nd epoch. The

training and validation accuracy can be seen in Fig. 6. Looking at the experimentation with the ResNet152 model, the data was used in running 100 epoch as done with the VGG16 model, passing through the model with our own data and displaying a step-by-step output for each epoch as well. The model was able to reach a 99% validation accuracy at the 20th epoch, whereas the highest accuracy was reached at the 95th epoch with accuracy of 99.65%. An overview of the accuracy of the model can be seen in Fig. 7. In Table 2, it can also be noted how the VGG16 and the ResNet152 models training progressed, showing the accuracy provided every 20 epochs, which demonstrates how the ResNet152 models stays slightly ahead of the VGG16 model in every iteration of training.

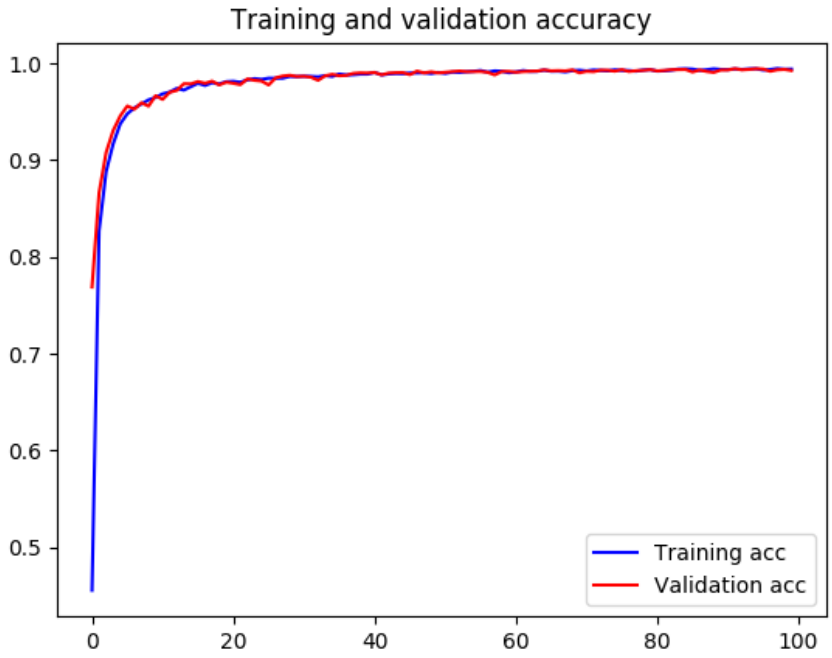

Fig. 6. The VGG16 model training and validation accuracy progress

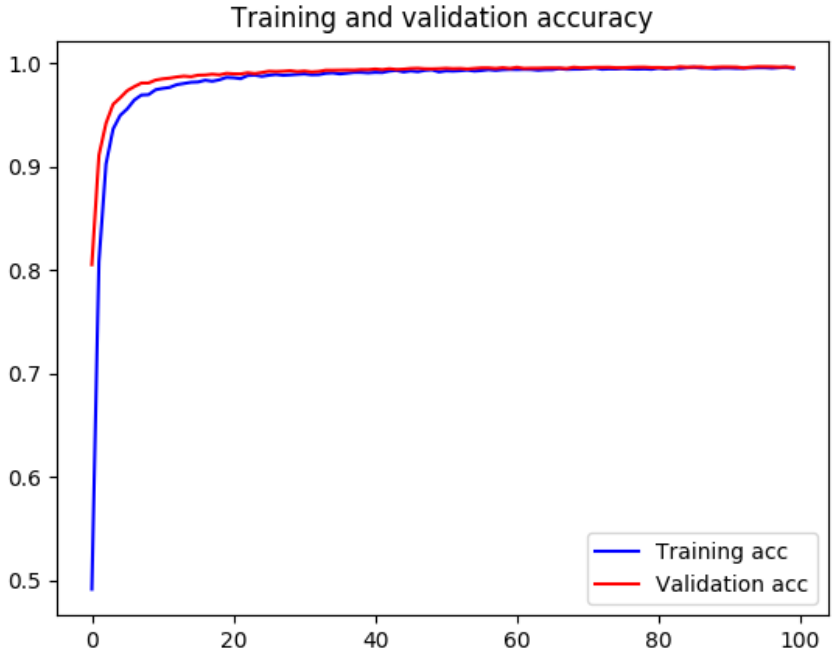

Fig. 7. The ResNet152 model training and validation accuracy progress

Table 2. Vgg16 and ResNet152 Progression Comparison

Epoch	VGG16 Accuracy	ResNet152 Accuracy
1	76.89%	80.51%
20	98.05%	99.00%
40	99.00%	99.36%
60	99.16%	99.48%
80	99.30%	99.56%
100	99.26%	99.57%

5 Conclusion

This paper presented a model utilizing fine-tuning with deep learning for the specific task of recognizing Arabic Sign Language, hoping to improve areas of related research such as sign language to sound techniques, translation of Arabic sign language to other languages, and many other areas.

The presented a model is an example on the application of CNN in image recognition and classification while reducing the size of the dataset required for training, and at the same time reaching higher accuracy. The work presented here begins by using state of the art network models such as the VGG-16, and Resnet152 that are already pre-trained,

and then apply fine-tuning using the ArASL dataset. Random under-sampling was applied to the dataset to reduce the imbalance resulting from the inconsistency of the class sizes, thus reducing the overall size of images from 54,049 to 25,600. The resulting model was cable of reaching a validation accuracy of 99.4% for the VGG 16 and 99.6% for the Resnet152.

In summary, the approach used not only had shown the great potential of using Arabic sign language imaging for classification, but also introduced fine-tuning to these images to remove the need of gathering a large dataset of images for training. Examining the field of Arabic Sign Language through the use of deep learning techniques, it can be determined that the work proposed here can be considered very novel in regards to the fact that in the Arabic Sign Language research field, no work can be found that involved the fine-tuning of well-known Convolutional Neural Networks for the purpose of Arabic Sign Language recognition. Moreover, the results provided have displayed great accuracy in recognition, therefore, can be dependable for more applicable use.

6 References

[1] W. H. Organization, "Deafness and hearing loss," 2019. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss. [Accessed: 28-Oct-2019].

[2] S. Halawani, "Arabic Sign Language Translation System on Mobile Devices," IJCSNS Int. J. Comput. Sci. Netw. Secur., 2008.

[3] G. A. Rao, K. Syamala, P. V. V Kishore, and A. Sastry, "Deep convolutional neural networks for sign language recognition," in *2018 Conference on Signal Processing and Communication Engineering Systems (SPACES)*, 2018, pp. 194–197.

[4] B. S. Parton, "Sign Language Recognition and Translation: A Multidisciplined Approach from the Field of Artificial Intelligence," *J. Deaf Stud. Deaf Educ.*, vol. 11, no. 1, pp. 94–101, 2005. https://doi.org/10.1093/deafed/enj003

[5] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv Prepr. arXiv1409.1556*, 2014.

[6] N. Tajbakhsh et al., "Convolutional Neural Networks for Medical Image Analysis: Full Training or Fine Tuning?" IEEE Trans. Med. Imaging, 2016.

[7] O. Russakovsky et al., "ImageNet Large Scale Visual Recognition Challenge," Int. J. Comput. Vis., 2015.

[8] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," arXiv Prepr. arXiv1602.07261, 2016.

[9] D. Singh and P. Garzon, "Using Convolutional Neural Networks and Transfer Learning to Perform Yelp Restaurant Photo Classification."

[10] G. Latif, N. Mohammad, J. Alghazo, R. AlKhalaf, and R. AlKhalaf, "ArASL: Arabic Alphabets Sign Language Dataset," Data Br., vol. 23, p. 103777, 2019. https://doi.org/10.1016/j.dib.2019.103777

[11] I. Arel, D. C. Rose, and T. P. Karnowski, "Deep Machine Learning - A New Frontier in Artificial Intelligence Research," IEEE Comput. Intell. Mag., 2010. https://doi.org/10.1109/mci.2010.938364

[12] J. Singha and K. Das, "Recognition of Indian Sign Language in Live Video," Int. J. Comput. Appl., vol. 70, no. 19, pp. 17–22, 2013. https://doi.org/10.5120/12174-7306

[13] M. Nachamai, "Alphabet recognition of american sign language: a hand gesture recognition approach using sift algorithm," Int. J. Artif. Intell. Appl., vol. 4, no. 1, p. 105, 2013. https://doi.org/10.5121/ijaia.2013.4108
[14] A. A. A. Youssif, A. E. Aboutabl, and H. H. Ali, "Arabic sign language (arsl) recognition system using hmm," Int. J. Adv. Comput. Sci. Appl., vol. 2, no. 11, 2011.
[15] M. Abdo, A. Hamdy, S. Salem, and E. M. Saad, "Arabic alphabet and numbers sign language recognition," Int. J. Adv. Comput. Sci. Appl., vol. 6, no. 11, pp. 209–214, 2015.
[16] N. El-Bendary, H. M. Zawbaa, M. S. Daoud, A. E. Hassanien, and K. Nakamatsu, "Arslat: Arabic sign language alphabets translator," in 2010 International Conference on Computer Information Systems and Industrial Management Applications (CISIM), 2010, pp. 590–595. https://doi.org/10.1109/cisim.2010.5643519
[17] M. A. Almasre and H. Al-Nuaim, "A Real-Time Letter Recognition Model for Arabic Sign Language Using Kinect and Leap Motion Controller v2," Int. J. Adv. Eng. Manag. Sci., vol. 2, no. 5, 2016.
[18] C.-H. Chuan, E. Regina, and C. Guardino, "American sign language recognition using leap motion sensor," in 2014 13th International Conference on Machine Learning and Applications, 2014, pp. 541–544. https://doi.org/10.1109/icmla.2014.110
[19] M. ElBadawy, A. S. Elons, H. Sheded, and M. F. Tolba, "A proposed hybrid sensor architecture for arabic sign language recognition," in Intelligent Systems' 2014, Springer, 2015, pp. 721–730. https://doi.org/10.1007/978-3-319-11310-4_63
[20] B. Kang, S. Tripathi, and T. Q. Nguyen, "Real-time sign language fingerspelling recognition using convolutional neural networks from depth map," in 2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR), 2015, pp. 136–140. https://doi.org/10.1109/acpr.2015.7486481
[21] L. Pigou, S. Dieleman, P.-J. Kindermans, and B. Schrauwen, "Sign language recognition using convolutional neural networks," in European Conference on Computer Vision, 2014, pp. 572–578. https://doi.org/10.1007/978-3-319-16178-5_40
[22] N. Priyadharsini and N. Rajeswari, "Sign Language Recognition Using Convolutional Neural Networks," Int. J. Recent Innov. Trends Comput. Commun., vol. 5, no. 6, pp. 625–628, 2017.
[23] H. Shin, W. J. Kim, and K. Jang, "Korean sign language recognition based on image and convolution neural network," in Proceedings of the 2nd International Conference on Image and Graphics Processing, 2019, pp. 52–55. https://doi.org/10.1145/3313950.3313967
[24] M. ElBadawy, A. S. Elons, H. A. Shedeed, and M. F. Tolba, "Arabic sign language recognition with 3d convolutional neural networks," in 2017 Eighth International Conference on Intelligent Computing and Information Systems (ICICIS), 2017, pp. 66–71. https://doi.org/10.1109/intelcis.2017.8260028
[25] S. Hayani, M. Benaddy, O. El Meslouhi, and M. Kardouchi, "Arab Sign language Recognition with Convolutional Neural Networks," in 2019 International Conference of Computer Science and Renewable Energies (ICCSRE), 2019, pp. 1–4. https://doi.org/10.1109/iccsre.2019.8807586
[26] S. Yang and Q. Zhu, "Continuous Chinese sign language recognition with CNN-LSTM," in Ninth International Conference on Digital Image Processing (ICDIP 2017), 2017, vol. 10420, p. 104200F. https://doi.org/10.1117/12.2281671
[27] A. K. Reyes, J. C. Caicedo, and J. E. Camargo, "Fine-tuning deep convolutional networks for plant recognition," in CEUR Workshop Proceedings, 2015.
[28] K. Yanai and Y. Kawano, "Food image recognition using deep convolutional network with pre-training and fine-tuning," in 2015 IEEE International Conference on Multimedia and Expo Workshops, ICMEW 2015, 2015. https://doi.org/10.1109/icmew.2015.7169816

[29] M. A. Hossen, A. Govindaiah, S. Sultana, and A. Bhuiyan, "Bengali Sign Language Recognition Using Deep Convolutional Neural Network," in 2018 Joint 7th International Conference on Informatics, Electronics & Vision (ICIEV) and 2018 2nd International Conference on Imaging, Vision & Pattern Recognition (icIVPR), 2018, pp. 369–373. https://doi.org/10.1109/iciev.2018.8640962

[30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778. https://doi.org/10.1109/cvpr.2016.90

[31] C. Shorten and T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," J. Big Data, 2019. https://doi.org/10.1186/s40537-019-0197-0

7 Authors

Yaser Saleh is a faculty of information technology and an Assistant Professor at the University of Petra in Jordan. He holds a Ph.D. degree in Computer Science from Loughborough University. His research interests include Artificial Intelligence, Deep learning and Image processing.

Ghassan Issa is a faculty of information technology and a Professor of Computer Science at the University of Petra in Amman, Jordan. He received his BET from Toledo University, Ohio, BSEE from Trine University, Indiana, M.S and PhD. from Old Dominion University, Virginia. His research interests include the areas in Artificial Intelligence, Machine learning, and e-Learning System. GIssa@uop.edu.jo

Article submitted 2020-01-08. Resubmitted 2020-03-01. Final acceptance 2020-03-05. Final version published as submitted by the authors.