Automated Arabic Sign Language Recognition System Based on Deep Transfer Learning

A.I. Shahin[1] **and Sultan Almotairi**[2,*]
[1]Department of Biomedical Engineering, HTI, Egypt.
[2]Department of Natural and Applied Sciences, Community College,
Majmaah University, Al- Majmaah 11952, Saudi Arabia

Summary
World Health Organization has reported that about 5% of the world's population is hearing-impaired. Automated sign language recognition system interfaces are classified into direct-device, vision-based and hybrid-based interfaces. Deep learning methodologies have been proven as an excellent tool for several automated computer vision systems. Moreover, deep learning overcame several difficulties existed inside traditional computer vision systems, A crucial need is found to provide deaf people with easy deep learning methods to interact with other people. In this paper, we propose a robust recognition system for Arabic sign language based on deep transfer learning. We employ transfer learning based on fine-tuning of existed pre-trained networks. In addition, we employ the data-augmentation to avoid overfitting and increase overall system performance. Several networks architectures have been examined for our target recognition task. We have also investigated the performance of residual networks versus plain networks. During our experiments, we employed Arabic sign language (ArSL2018) public dataset that consists of 54,049 images with 32 class. The overall system accuracy achieved by the proposed one is 99.52% and 99.5% sensitivity based on ResNet18 Architecture with data augmentation benefits. A powerful Arabic sign language recognition system based on deep learning theory is proposed which can be employed later in several automated sign language recognition tasks.

Key words:
Arabic Sign Language, Deep Learning, Convolutional Neural Network Transfer Deep Learning, Residual Networks.

1. Introduction

A sign language science consists of a set of hand gestures which enable a person to express letters, words, and expressions of a specific language. Systems that can recognize sign-language gestures can be employed to make the communication easy between the hearing-impaired and other people. Moreover, human-machine interaction systems can largely benefit from advances in sign language recognition (SLR). Creation of visual Sign language datasets is very important to construct automated systems. Such datasets were created in relation to different regions and spoken language such as (Arabic, English, Germany, Indian…etc.) and sign language application area such as (isolated recognition continuous recognition tracking translation) [1-5]. Human-computer interaction acquired a growing interest, especially in SLR systems. Automated SLR systems can be categorized into 3 approaches: sensor-based approach, vision-based approach, and hybrid-based approach [6]. The sensor-based approach utilizes measurable readings that are in direct contact with a human hand such as flexion sensors, instrumented gloves, and position-tracking devices. The vision-based approach captures the sequence movements of human hands using a camera. The hybrid approach combines the two previous approaches in one system to increase system performance. However, the complexity of such systems increases with hardware sensors integration. Vision-based systems advantage is that the low installation requirements due to any attached devices. On the other hand, their main disadvantage is that they require a complex computation for hand position extraction before performing any recognition on the acquired images.

Computer vision systems employed traditional machine learning approaches which consist of pre-processing, segmentation, features extraction, detection, and recognition [7]. These sequentially steps in tracking vision tasks consume a lot of time. Moreover, such systems were trained on limited size datasets which leak to robustness and generalization.

Deep learning approach in computer vision applications has been proven as an excellent tool for computer vision tasks [8-9]. Deep learning methodologies have been invoked in several computer vision tasks such as pre-processing [10], enhancement [11], segmentation [12], detection [13], and recognition [14]. Deep learning has been contributed to increasing several recognition system performances using several network architectures based on supervised, semi-supervised, and unsupervised learning [15]. Supervised learning based on convolutional neural network (CNN) is one of the most successful tools employed for different visual recognition tasks. Moreover, unlike the traditional machine learning algorithms, CNN can be generalized, transfer learning and fine-tuning its learned parameters which have been acquired from other visual recognition tasks.

Manuscript received October 5, 2019
Manuscript revised October 20, 2019

Fig.1 Representation of Arabic sign language characters [16].

Deeply speaking, sign language is mapped to manual signs (MS) and non-manual signs (NMS). MS consist of static hand gestures, fingerspelling, and hand motions. NMS consist of facial expressions, lip-reading, and body language. Previous Arabic SLR systems worked on separated characters and numbers [16], words [18], and continuous words [18] for Arabic language. As shown in Fig 1, a sample of separated 32 Arabic characters which were prepared through ArASL dataset [16].

This paper introduces a system which can recognize the Arabic sign language based on vision-based approach. Arabic sign language dataset was employed to train the proposed system. Deep learning methodologies are employed to achieve the highest accuracy for the proposed system. We investigate different deep network learning methods based on CNN. Obtained results are promising with no need for gloves and sensors.

This paper is organized as follows. Firstly, we introduce a background for the literature on Arabic SLR systems. Secondly, we describe our proposed method and its main phases. Finally, we present different experimental results for each step.

2. Related Work

Many recent research articles were introduced for SLR in the literature. Here, a brief summary of the previous Arabic SLR systems is given in Table.1. In [19-22], SLR systems were proposed for Arabic isolated word recognition. It is noticed that all continuous recognition systems have limited dataset size. However, the dataset size in isolated recognition systems reaches to 2323 sample as in [20]. It is also noticed that the highest performance was achieved in [21], [22] recognition systems with 99.5 % accuracy score value. Such systems cannot be generalized according to small dataset size. The only system based on deep belief network [24] was only applied for 200 samples which caused low system performance reached to 85%. On the other hand, the deep belief has not been employed for isolated Arabic alphabet. All systems in [19-23], [25-26] employed traditional classifiers such as heuristic approach, KNN, neural network, recurrent neural network, neuro-fuzzy classifier, and support vector machine classifier. These previous systems employed image-based features like intensity features, morphological features, and 2D discrete Fourier transform features. In [22-23], a hybrid approach based on visual recognition for specified colored gloves was proposed. However, this approach is not applicable in real life with lighting conditions limitations.

Deep learning has been widely used for different sign languages [27]. However, there are a few of researches that applied deep learning for Arabic sign language recognition [24]. In [26], 3D CNN is designed by adding the third dimension as a motion dimension. The 3D CNN is suitable for continuous word recognition where 3D kernel is convolving over multiple consequence frames. In [28], Bangla sign language recognition system based on CNN network which achieved 90 % accuracy for only 10 digits Bangla alphabet. In [29], Italian sign language recognition system based on CNN network was proposed which achieved 91 % accuracy for 20-digit Italian alphabet. In [30], American Sign Language sign language recognition system based on CNN network was proposed. Both systems accuracies varied from achieved 82.5 to 90 % for 26-digits American alphabet. In [31], a hybrid deep learning methodology based on CNN and recurrent neural network were proposed to recognize American sign language which achieved 91% accuracy. In [32], Indian sign language recognition system based on CNN network was proposed which achieved 92 % accuracy for 46-digit Indian alphabet.

In this paper, our contributions are as follows. A robust recognition system is proposed for a specific sign language application in Arabic, which employs transfer deep learning methods to process large dataset. Usage of large size dataset source - acquired under different lighting conditions make the system more adaptive and the classification problem to be more difficult. ArSl2018 dataset is the first time to be examined through deep learning framework. We have also compared between several pre-trained architectures either plain or residual networks. We investigate the data-augmentation effect as a pre-processing step in our proposed system.

Table 1: Survey of recent Arabic SLR systems.

Ref.	Dataset Name	Dataset Type	Dataset Size	Classifier	System Performance
[19]	Signs World Atlas	Isolated/ Continues	67 images for Arabic alphabet.	Heuristic Classifier	ACC=95%
[20]	ArSL dataset	Isolated	2323 sample for Arabic alphabet.	Polynomial Classifier	Error rate= 6.59%
[21]	ArASLRDB	Isolated	357 images for Arabic alphabet and numbers.	HMM Classifier	ACC=99 %
[22]	Arabic manual alphabets	Isolated	1800 images for Arabic alphabet	Neuro-Fuzzy Classifier	ACC=93%
[23]	Arabic manual alphabets	Isolated/ Word	900 images for Arabic alphabet	Recurrent –neural network Classifier	ACC=95.5%
[24]	Arabic sign language dictionary	Continues	200 sample for Arabic alphabet.	Deep belief network	ACC=85 %
[25]	LAS: Second part of the Unified Arabic Sign Dictionary	Continues	100 sample for Arabic alphabet.	ensemble Subspace KNN	ACC=81%
[26]	ArSL Database	Continues / Word	150 sample for 23 words for Arabic alphabet.	Support vector machine	ACC=99.5%

3. Proposed method

Deep learning methodologies have been established as a generalization concept. Transfer learning approach has been proved as an excellent tool for sign language recognition tasks [26-32]. Based on pre-trained networks, we can employ these learned networks on several image recognition tasks. One of the most challenges here is how to employ such learned networks for identifying the different sign language classes using Arabic sign language dataset. Traditionally, pre-trained networks have been trained on naturally images datasets such as CIFAR10 / CIFAR100 [33], Caltech 101/ Caltech 256 [34], ImageNet [35]. Each network trained on such datasets constructs learned parameters beside its own architecture. In our proposed system, we employ different pre-trained networks to extract deep features and to process them to perform recognition step. The proposed system shown in Fig.2, consists of pre-processing, re-train pre-trained network, deep features extraction, and classification.

3.1 Pre-processing

The pre-processing stage consists of two procedures; firstly, scaling input images. Secondly, image data-augmentation.

Scaling input images is done by resizing these images to fit each pre-trained CNNs image input layer. On the other hand, each of such networks has its own input layer size: AlexNet and SqueezeNet image input layer are (227×227×3). VGGNet16/19, GoogleNet, DenseNet, MobileNet and ResNet 18/50/101 image input layer is (224×224×3). InceptionV3 image input layer is (299×299×3). Since ArSL2018 images dataset are collected at (64×64) resolution, the images should be exposed to up-sampling according to resolution.

Data augmentation has been introduced in [36] by eight types; flips, Gaussian noise, jittering, scaling, powers, gaussian blur, rotations, and shears. In this paper, we employ rotation and translation in both x and y-direction. The rotation transformation matrix is given in Eq.1

$$Ar = \begin{pmatrix} cos\theta & sin\theta & 0 \\ -sin\theta & cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (1)$$

where Ar is the rotation transformation matrix and θ is the rotation angle. The MATLAB data augmenter function provides randomly rotation angles from a continuous uniform distribution within the specified interval. In this paper, we used -20, +20 rotation angles.

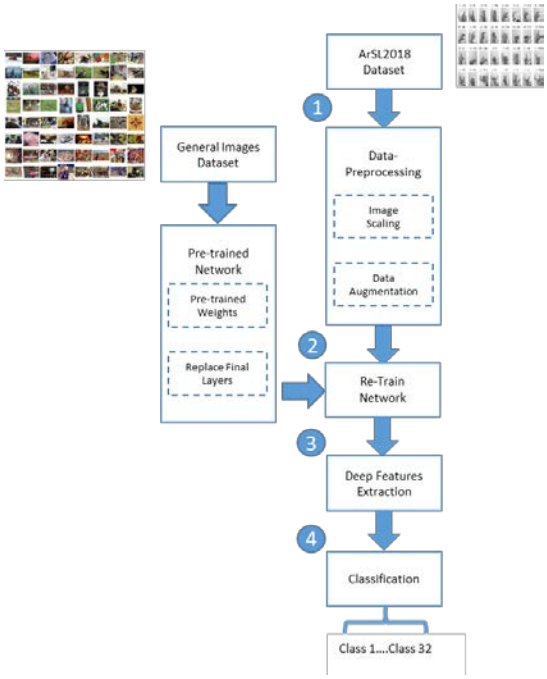

Fig. 2 Proposed system workflow

Translation just involves moving the image along the X or Y direction (or both). Translation benefit is to make CNN

attention to look at objects that can be located at almost anywhere in the image. The translation transformation matrix is defined in Eq. 2. In this paper, we used -3, +3 translation values in both *x* and *y*.

$$At = \begin{pmatrix} 1 & 0 & tx \\ 0 & 1 & ty \\ 0 & 0 & 1 \end{pmatrix} \quad (2)$$

where *At* is the transformation translation matrix and *tx*,*ty* is the translation values in *x, y*-direction.

3.2 Re-train Pre-trained Networks

Pre-trained networks can be employed through two methodologies, the first is based on freezing the first pre-trained layers and the second is based on re-training the network again without keeping the initial layers. The decision refers to the nature of images dataset [37]. In this paper, as the sign language dataset nature differs from the general images' datasets. We perform re-training for the pre-trained networks. In Eq.3, gradient descent can be defined as:

$$W_{n+1} = W_n + \gamma \nabla F(W_N) \quad (3)$$

where *W* represents CNN learned parameters through the training process, γ is the movement distance from the activation function *F*, which takes the value of the previous parameters as input. One of the disadvantages of gradient descent (GD) method is that it uses all the training data to find the optimal parameters.in our paper, The training process is based on stochastic gradient descent (SGD) which is defined in equation (4) as SGD method training utilizes mini-batch units (Z of Eq.4) randomly selected from the overall training data [38]:

$$W_{n+1} = W_n + \gamma \nabla F(Z_n, W_n) \quad (4)$$

3.3 Deep Features Extraction

The deep features are extracted by processing the input images through different layers, which gives it the ability to automatically extract the leaned patterns from each image. These layers are as the following, convolution layer, pooling layer, batch normalization layer, Relu activation layer, dropout layer, and fully connected layer.

Convolutional layer is responsible for extracting specified similar patterns associated with a spatial area. A small size filters are applied to the input image with learnable weights. These weights are extracted during the training process. The convolution process is explained in Eq.5.

$$F_I^K = I_{x,y} * K_I^k \quad (5)$$

where $I_{x,y}$ represent input image (x, y), K_I^k represents I^{th} convolutional kernel of k^{th} layer.

The pooling layer is utilized to down sample the high dimension learnable parameters which reflect on decreasing computational cost as defined in Eq.6:

$$Z_I = f_p(F_{x,y}^I) \quad (6)$$

where Z_I represents I^{th} output feature map, $F_{x,y}^I$ represents I^{th} input feature map, and f_p represents pooling operation type.

The activation function is responsible for taking the decision in the neural network. The traditional activation functions used in neural networks are sigmoid and tanh. To accelerate the processing time and increase the recognition power for CNN architectures [39], several activations functions were developed such as ReLU [40] such as leaky ReLU [41], ELU [42], and PReLU [43].

$$T_I^k = f_A(F_I^k) \quad (7)$$

where f_A represents the activation and F_I^k represent the output of convolution layer operation.

Batch normalization helps to increase CNN stabilization through different CNN layers. Each activation layer output is normalized by subtracting the average of each batch, then divide it by standard deviation of each batch. The deep network training can be accelerated by reducing internal covariate shift. As in Eq.7, we can define the batch normalization N_I^k of a given feature map T_I^k with standard variation σ.

$$N_I^k = \frac{T_I^k}{\sigma^2 + \sum_i T_i^k} \quad (8)$$

Dropout layer helps to regularize CNN output. In dropping out process, some of CNN neurons are dropped out stochastically during training process. This operation helps to decrease over-fitting.

Fully connected (FC) layer is placed on the last CNN architecture. FC layer re-map the learned high dimension features into single dimension form. In the fine-tuning process, the first FC layer is almost defined with 1000 class. So, we perform fine-tuning for the FC layer and replace final layers to 32 class only for Arabic sign languages dataset.

In our proposed method, we employ different deep neural network architectures as shown in Fig.3. In-state of art CNN introduced by LeCun et al. [44], the plain CNN is used to extract deep features as shown in Fig3 (a). However, in other pre-trained networks, the residual CNN network is used to extract features as shown in Fig. 3(b).

Residual network was proposed by He et al. while introducing ResNet architectures [45]. Moreover, ResNet was deeper than other plain networks such as LeNet, AlexNet, and VGGNet. In the previous studies [46-47], residual block helps to increases the recognition performance of plain network. In this paper, we investigate

the performance of residual networks with sign language recognition task.

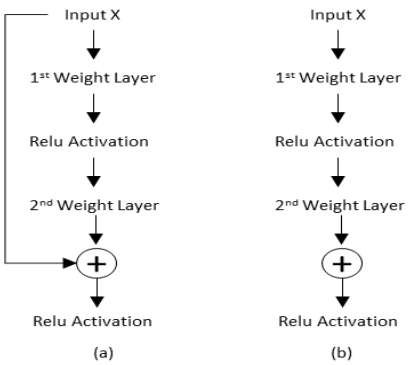

Fig. 3 Deep network architectures: (a) residual network, (b) plain network.

3.4 Classification

Softmax function [48] is used to classify the target classes. Softmax function determines a of normalized probability score for each class. The Softmax function is defined in Eq.5.

$$f_i(z) = \frac{e^{z_j}}{\sum_k e^{z_k}} \tag{9}$$

The cross-entropy loss [48] is used to obtain class scores of which is formulated in Eq.6.

$$L_i = f_{y_i} + log \sum_j e^{f_j} \tag{10}$$

where the f_j refers to the j_{th} element of the vector of class scores *f*.

4. Experimental Results

In this paper, a public published dataset called ArSL2018 dataset is utilized [16]. The dataset contains Arabic sign language letter with 32 characters and saved in JPG images with 8 bits depths. The total no. of images are 54,049 images. The dataset was acquired from different people with different lighting, angles, and background. The dataset was randomly divided into 90% training set and 10% testing set. During our experiments, we implement our code using the MATLAB 2019a. The system platform containing Quad-Core 2.9 GHz Intel i5 with 16GB RAM. The GPU computation is done through NVIDIA TITAN-Xp with 12 GB RAM.

The proposed sign language recognition system was evaluated using metrics followed in [49]. We employ the confusion matrix and its influenced metrics as in Eq.7 to Eq.13 :

$$Accuracy = \frac{TP+TN+FP+FN}{TP+FN} \tag{11}$$

$$Error = 1 - Accuracy \tag{12}$$

$$Sensitivity = \frac{TP}{TP+FN} \tag{13}$$

$$Specifity = \frac{TN}{TN+FP} \tag{14}$$

$$F1\,Score = \frac{2TP}{2TP+FP+FN} \tag{15}$$

$$\text{Matthews correlation coefficient (MCC)} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \tag{16}$$

$$\text{Kappa} = 1 - \frac{1-P_0}{1-P_e} \tag{17}$$

The experimental results are introduced into two sub-sections: training and testing results. We investigate the performance of several pre-trained networks during both training and testing results. Finally, data augmentation samples will be introduced.

4.1 Training Results

For training process, we utilized SGD optimizer with the following setting: 60 epochs, min-batch size 128 which generate 45600 iterations during the training process as followed in [50]. In Fig. 4, the learning curves of accuracy for all state of art CNN networks is very low. On the other hand, all residual deep networks perform well with a high learning curve. It is also noticed that the highest depth network achieved well learning curve. On the other hand, In Fig. 5, the learning loss curves for all state of art CNN networks is very high compared with all residual deep networks. It is also noticed that the highest depth network achieved well learning curve.

Execution time for training each network on ArSL2018 dataset are reported in Fig.6 with the help of the parallel computation of a GPU. The most important limitation of using DenseNet is the training cost time which is about 41 hours. DenseNet training cost time is too much related to other pre-trained networks. This can be explained by its highest depth in opposite to other pre-trained networks which reach to 709 layers. On the other hand, plain CNN networks achieved such as AlexNet, VGGNet16, and VGGNet19 were varying in training cost time between 3.14 h for AlexNet, 20 h for VGGNet16, and 23 h for VGGNet19. This can be explained also by the differentiation in each network depth and no. of layers. Different ResNet architectures are also varied in training time due to each network layers numbers. ResNet18

achieved the lowest training cost time 4.3h, and ResNet101 achieved the highest training cost time with 24h. GoogleNet achieved reasonable training cost time with its high layers' number with 6h.

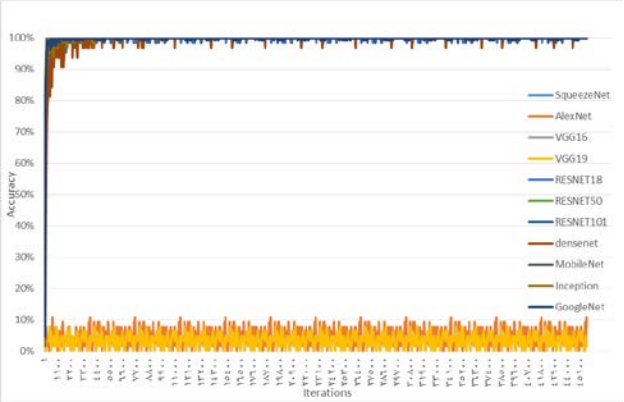

Fig. 4 Training accuracy vs. iterations number through different pre-trained networks.

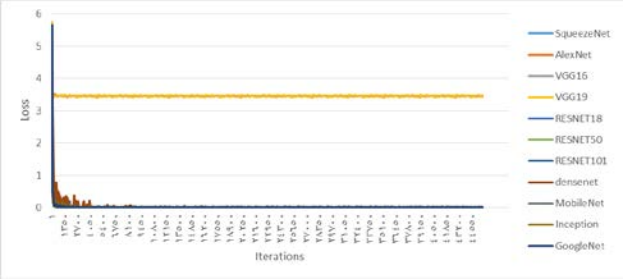

Fig. 5 Training loss vs. iterations number through different pre-trained networks.

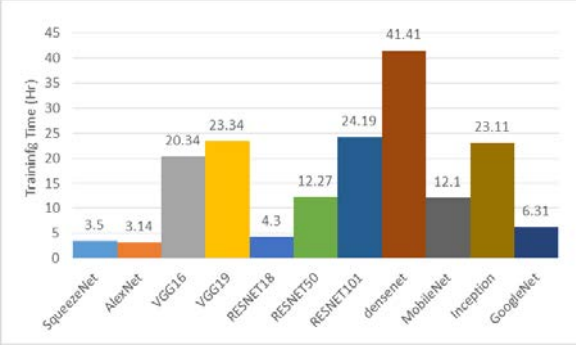

Fig. 6 Training time required for each pre-trained network.

4.2 Training Results

Firstly, we demonstrate the testing results through the most important evaluation parameter which is the accuracy through different pre-trained networks. Secondly, we demonstrate accuracy, error, F1 Score, sensitivity, precision, specificity, MCC, kappa for all pre-trained network. Thirdly, we demonstrate the confusion matrices for the highest performance model.

Finally, we visualize data-augmentation results with its effect on enhancing the recognition performance.

Fig.7 demonstrates that the poor performance of plain networks with accuracy score of 3.91%. However, the performance of pre-trained residual networks exists in a positive correlation to the depth of the networks in ArSL2018 dataset described in the previous section. These results give obvious evidence of the central importance of residual networks and its superior performance over the plain networks in Arabic sign language recognition systems. The accuracy of ResNet101 reaches 99.52%.

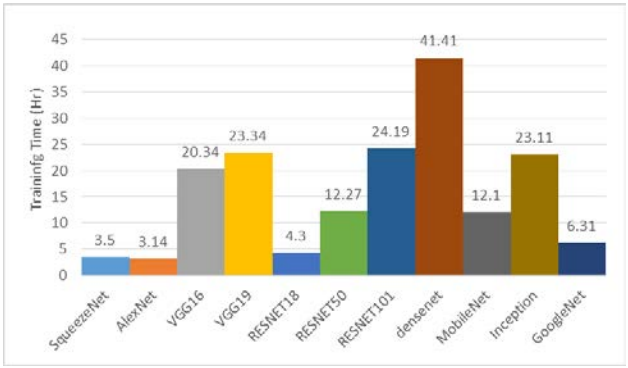

Fig. 7 Training time required for each pre-trained network.

Table.2 demonstrates that the performance of transfer learning approach based on different pre-trained networks. The dataset has been used in training each network without data-augmentation. It is noticed that the highest performance of residual networks with its different architectures, depth, and no. of layers. ResNet 18 achieved the highest specificity 100 %. The highest sensitivity was achieved by all ResNet networks 18/50/101 with 99.5%. On the other hand, the highest accuracy value achieved by ResNet101 network with a minimum error of 0.48%. The highest error percentage achieved by all plain networks. This proves that the superior performance of residual networks in Arabic sign language recognition. However, SqueezeNet network performs better than AlexNet in the previous recognition tasks. The performance of SqueezeNet was similar to AlexNet, VGGNet 16/19 with our recognition task.

In Table.3, the data augmentation procedure effect has been investigated to increase the performance of ResNet18 from accuracy score of 99.48% to 99.52%. Besides, the highest achieved specificity value reached to 100% and precision reached 99.5%. In Fig. 9, a sample of the augmented dataset is displayed using rotation and translation techniques that have been introduced inside the proposed system section.

Table 2- Recognition system evaluation parameters through different pre-trained networks.

	Error	Sensitivity	Specificity	Precision	FPR	F1 score	MCC	Kappa
AlexNet	0.9609	0.0313	0.9688	0	0.0313	0	0	0.937
VGG16	0.9609	0.0313	0.9688	0	0.0313	0	0	0.937
VGG19	0.9609	0.0313	0.9688	0	0.0313	0	0	0.937
SqueezeNet	0.9609	0.0313	0.9688	0	0.0313	0	0	0.937
DenseNet	0.005	0.995	0.9998	9.95E-01	1.61E-04	0.9951	0.9949	0.9174
InceptionV3	5.00E-03	0.9949	0.9998	0.9953	1.62E-04	0.9951	0.9949	0.9174
MobileNet	0.0054	0.9946	0.9998	0.9947	1.73E-04	0.9946	0.9945	0.9113
Resnet18	0.0052	0.995	1	0.995	1.67E-04	0.9948	0.9946	0.9144
Resnet50	0.0049	0.995	0.9998	0.995	1.61E-04	0.995	0.995	0.9174
Resnet101	0.0048	0.995	0.9998	0.9953	1.55E-04	0.9952	0.995	0.9205

Table 3- Recognition system evaluation parameters for ResNet18 before and after data augmentation.

	Accuracy	Error	Sensitivity	Specificity	Precision	FPR	F1 score	MCC	Kappa
Resnet18 without Data Augmentation	0.9948	0.0052	0.995	1	0.995	1.67E-04	0.9948	0.9946	0.9144
Resnet18 with Data Augmentation	0.9952	0.0048	0.995	1	0.9955	1.55E-04	0.9952	0.995	0.9205

ArSL2018 dataset consists of 32 Arabic sign language class. We used the confusion matrix chart as shown in Fig.8 between the true labels and the predicted labels. We specify a normalized row to visualize the true positive rates and false-positive rate. Also, we specify normalized columns to visualize the positive predictive values and false discovery rates in the column summary. Character 'dha' achieved the highest error rate with 6 interferences with class 'ta'. This can be explained by their high similarity in the image. 16-characters have been clearly recognized with no confusion found which are "Yaa", "toot", "taa", "seen", " ra", "laam" , " la", " kaaf", "jeem", "ha", "ghain", "dhad", "aleff", "bb", and "ain". The remind character percentage error are almost 1 or two error.

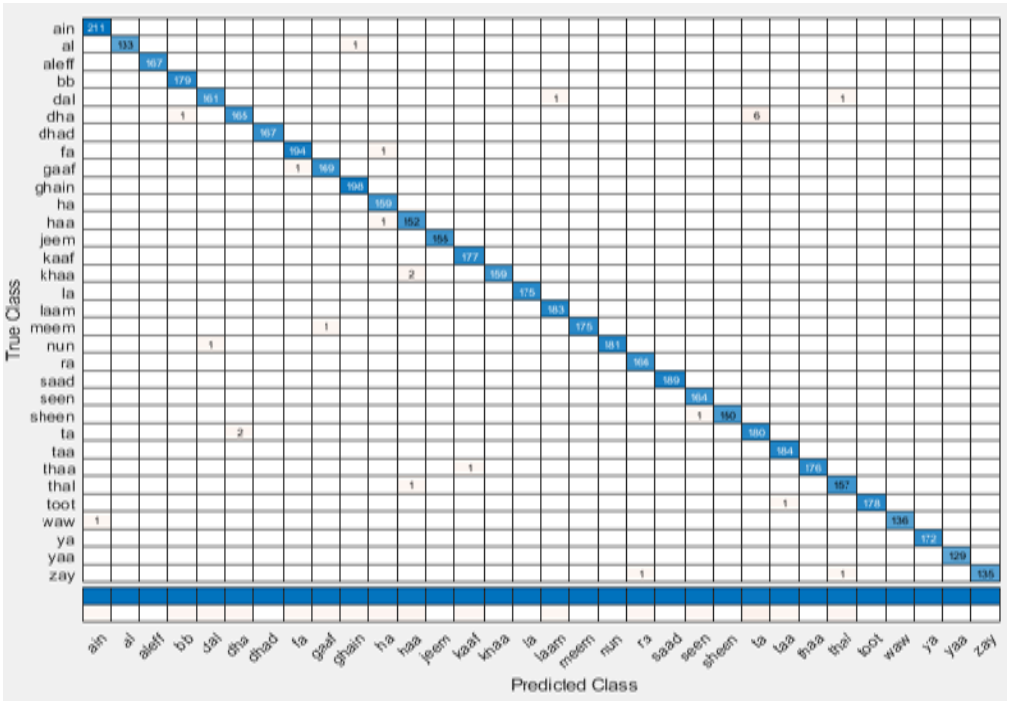

Fig. 8 Confusion matrix result of best performance ResNet architecture.

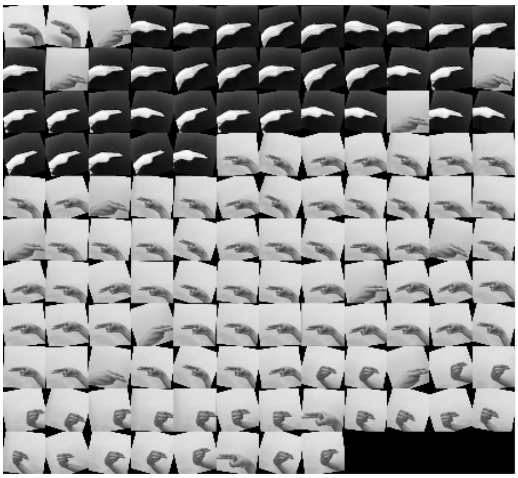

Fig. 9 Data augmentation sample results.

5. Conclusion

This paper proposed a new ASLR method by using transfer learning approach of deep convolutional neural network. The proposed system based on residual network ResNet101 achieved the highest accuracy score with 99.52%. On the other hand, the plain CNN network achieved very low accuracy score with 3.9%. Applying data augmentation procedure achieved a reasonable improvement in ResNet18 performance with 0.04% to reach ResNet101 accuracy. The proposed system based on ResNet18 achieved highest performance 99.52% accuracy, 100% specificity, and 99.5% precision reached with low training cost time reached to 4h. The experimental results are very promising for applying CNN in Arabic sign language recognition. Our approach performance can be increased with using residual CNN and data augmentation process. There is a real need to construct a simple residual network to save training time and decrease computational complexity. The trained network can re-employed to extract deep features for other Arabic sign language recognition tasks. The most important limitation lies in the required training cost time, which can be tackled in the future work by proposing low depth residual network. The proposed system can be applied in a fully-automated system for Arabic sign language recognition system.

Acknowledgment

The author would like to thank Deanship of Scientific Research, Majmaah University (Grant no. R-1441-18) for funding this work. Also, the authors would like to thank LATIF, Ghazanfar, et al. [16] for providing their valuable dataset.

References

[1] ANDERSON, Ricky, et al. Sign language recognition application systems for deaf-mute people: A review based on input-process-output. Procedia computer science, 2017, 116: 441-448.

[2] MOHANDES, Mohamed; DERICHE, Mohamed; LIU, Junzhao. Image-based and sensor-based approaches to Arabic sign language recognition. IEEE transactions on human-machine systems, 2014, 44.4: 551-557.

[3] STARNER, Thad; WEAVER, Joshua; PENTLAND, Alex. Real-time american sign language recognition using desk and wearable computer based video. IEEE Transactions on pattern analysis and machine intelligence, 1998, 20.12: 1371-1375.

[4] RAJAM, P. Subha; BALAKRISHNAN, G. Real time Indian sign language recognition system to aid deaf-dumb people. In: 2011 IEEE 13th International Conference on Communication Technology. IEEE, 2011. p. 737-742.

[5] KOLLER, Oscar; FORSTER, Jens; NEY, Hermann. Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers. Computer Vision and Image Understanding, 2015, 141: 108-125.

[6] MOHANDES, Mohamed; DERICHE, Mohamed; LIU, Junzhao. Image-based and sensor-based approaches to Arabic sign language recognition. IEEE transactions on human-machine systems, 2014, 44.4: 551-557.

[7] SONKA, Milan; HLAVAC, Vaclav; BOYLE, Roger. Image processing, analysis, and machine vision. Cengage Learning, 2014.

[8] GUO, Yanming, et al. Deep learning for visual understanding: A review. Neurocomputing, 2016, 187: 27-48.

[9] NGUYEN, Van Nhan; JENSSEN, Robert; ROVERSO, Davide. Automatic autonomous vision-based power line inspection: A review of current status and the potential role of deep learning. International Journal of Electrical Power & Energy Systems, 2018, 99: 107-120.

[10] WATKINS, Yijing Z.; SAYEH, Mohammad R. Image data compression and noisy channel error correction using deep neural network. Procedia Computer Science, 2016, 95: 145-152.

[11] LIU, Yu, et al. Deep learning for pixel-level image fusion: Recent advances and future prospects. Information Fusion, 2018, 42: 158-173.

[12] Guo, Y., Liu, Y., Oerlemans, A., Lao, S., Wu, S., & Lew, M. S. (2016). Deep learning for visual understanding: A review. Neurocomputing, 187, 27-48.

[13] CRUZ-ROA, Angel Alfonso, et al. A deep learning architecture for image representation, visual interpretability and automated basal-cell carcinoma cancer detection. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, Berlin, Heidelberg, 2013. p. 403-410.

[14] JIANG, Xiaoheng, et al. Deep neural networks with elastic rectified linear units for object recognition. Neurocomputing, 2018, 275: 1132-1139.

[15] KIRAN, B.; THOMAS, Dilip; PARAKKAL, Ranjith. An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos. Journal of Imaging, 2018, 4.2: 36.

[16] LATIF, Ghazanfar, et al. ArASL: Arabic Alphabets Sign Language Dataset. Data in Brief, 2019, 23: 103777.
[17] SHOHIEB, Samaa M.; ELMINIR, Hamdy K.; RIAD, A. M. Atlas; a benchmark Arabic Sign. 2014.
[18] NEIVA, Davi Hirafuji; ZANCHETTIN, Cleber. Gesture recognition: a review focusing on sign language in a mobile context. Expert Systems with Applications, 2018, 103: 159-183.
[19] SHOHIEB, Samaa M.; ELMINIR, Hamdy K.; RIAD, A. M. Signsworld atlas; a benchmark Arabic sign language database. Journal of King Saud University-Computer and Information Sciences, 2015, 27.1: 68-76.
[20] ASSALEH, Khaled; AL-ROUSAN, M. Recognition of Arabic sign language alphabet using polynomial classifiers. EURASIP Journal on Advances in Signal Processing, 2005, 2005.13: 507614.
[21] ABDO, M., et al. Arabic alphabet and numbers sign language recognition. International Journal of Advanced Computer Science and Applications, 2015, 6.11: 209-214.
[22] AL-JARRAH, Omar; HALAWANI, Alaa. Recognition of gestures in Arabic sign language using neuro-fuzzy systems. Artificial Intelligence, 2001, 133.1-2: 117-138.?
[23] MARAQA, Manar; ABU-ZAITER, Raed. Recognition of Arabic Sign Language (ArSL) using recurrent neural networks. In: 2008 First International Conference on the Applications of Digital Information and Web Technologies (ICADIWT). IEEE, 2008. p. 478-481.
[24] HASASNEH, Nabil; HASASNEH, Ahmad; TAQATQA, Sameh. Towards Arabic Alphabet and Numbers Sign Language Recognition. 2017.
[25] SIDIG, Ala Addin I.; MAHMOUD, Sabri A. Trajectory based Arabic Sign Language Recognition. International Journal of Advanced Computer Science and Applications, 2018, 9.4: 283-291.
[26] ELBADAWY, Menna, et al. Arabic sign language recognition with 3d convolutional neural networks. In: 2017 Eighth International Conference on Intelligent Computing and Information Systems (ICICIS). IEEE, 2017. p. 66-71.
[27] ASADI-AGHBOLAGHI, Maryam, et al. A survey on deep learning based approaches for action and gesture recognition in image sequences. In: 2017 12th IEEE international conference on automatic face & gesture recognition (FG 2017). IEEE, 2017. p. 476-483.
[28] ISLAM, Sanzidul, et al. A Potent Model to Recognize Bangla Sign Language Digits Using Convolutional Neural Network. Procedia computer science, 2018, 143: 611-618.
[29] PIGOU, Lionel, et al. Sign language recognition using convolutional neural networks. In: European Conference on Computer Vision. Springer, Cham, 2014. p. 572-578.
[30] BHEDA, Vivek; RADPOUR, Dianna. Using deep convolutional networks for gesture recognition in American sign language. arXiv preprint arXiv:1710.06836, 2017.
[31] BANTUPALLI, Kshitij; XIE, Ying. American Sign Language Recognition using Deep Learning and Computer Vision. In: 2018 IEEE International Conference on Big Data (Big Data). IEEE, 2018. p. 4896-4899.
[32] RAO, G. Anantha, et al. Deep convolutional neural networks for sign language recognition. In: 2018 Conference on Signal Processing And Communication Engineering Systems (SPACES). IEEE, 2018. p. 194-197.
[33] Cs.utoronto.ca. (2019). CIFAR-10 and CIFAR-100 datasets. [online] Available at: http://www.cs.utoronto.ca/~kriz/cifar.html [Accessed 1 Aug. 2019].
[34] Vision.caltech.edu. (2019). Caltech256. [online] Available at: http://www.vision.caltech.edu/Image_Datasets/Caltech256 [Accessed 1 Sep. 2019].
[35] Image-net.org. (2019). ImageNet. [online] Available at: http://www.image-net.org/ [Accessed 1 Sep. 2019].
[36] SHORTEN, Connor; KHOSHGOFTAAR, Taghi M. A survey on Image Data Augmentation for Deep Learning. Journal of Big Data, 2019, 6.1: 60.
[37] JMOUR, Nadia; ZAYEN, Sehla; ABDELKRIM, Afef. Convolutional neural networks for image classification. In: 2018 International Conference on Advanced Systems and Electric Technologies (IC_ASET). IEEE, 2018. p. 397-402.
[38] SAINATH, Tara N., et al. Deep convolutional neural networks for large-scale speech tasks. Neural Networks, 2015, 64: 39-48.
[39] LECUN, Yann; BENGIO, Yoshua; HINTON, Geoffrey. Deep learning. nature, 2015, 521.7553: 436.
[40] GU, Jiuxiang, et al. Recent advances in convolutional neural networks. Pattern Recognition, 2018, 77: 354-377.
[41] WANG, Tao, et al. End-to-end text recognition with convolutional neural networks. In: Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012). IEEE, 2012. p. 3304-3308.
[42] XU, Bing, et al. Empirical evaluation of rectified activations in convolutional network. arXiv preprint arXiv:1505.00853, 2015.
[43] DALGLEISH, Tim, et al. Reduced specificity of autobiographical memory and depression: the role of executive control. Journal of Experimental Psychology: General, 2007, 136.1: 23.
[44] LECUN, Yann, et al. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 1998, 86.11: 2278-2324.
[45] HE, Kaiming, et al. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2016. p. 770-778.
[46] KRIZHEVSKY, Alex; SUTSKEVER, Ilya; HINTON, Geoffrey E. Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. 2012. p. 1097-1105.
[47] SIMONYAN, Karen; ZISSERMAN, Andrew. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.
[48] BISHOP, Christopher M. Pattern recognition and machine learning. springer, 2006.
[49] HOANG, Toan Minh, et al. Deep retinanet-based detection and classification of road markings by visible light camera sensors. Sensors, 2019, 19.2: 281.
[50] KOO, Ja, et al. CNN-Based Multimodal Human Recognition in Surveillance Environments. Sensors, 2018, 18.9: 3040.