Wide Residual Network for Vision-based Static Hand Gesture Recognition

Yong Soon Tan, Kian Ming Lim, *Member, IAENG,* and Chin Poo Lee, *Member, IAENG*

***Abstract*—Hand gesture is a communication tool that allows messages to be conveyed, actions to be performed through hand gestures. Hence, it has the ability to simplify communication and enhance human computer interaction. This paper proposed Wide Residual Network for static hand gesture recognition. WRN improves feature propagation and gradient flows by utilizing shortcut connection in residual block. Wide residual block further improves upon residual block by increasing the width of the network and improving feature reuse, and thereby allowing the depth of the network to be trimmed and fewer trainable parameters to be learned. The network is experimented on three public datasets and compared with existing convolutional neural network (CNN) variants proposed for static hand gesture recognition. Experimental results show Wide Residual Network outperforms the existing CNN variants proposed for hand gesture recognition.**

***Index Terms*—Hand gesture recognition, Sign language recognition, Convolutional Neural Network (CNN), Wide Residual Network**

I. Introduction

HAND gesture serves as an alternative communication tool, but for certain communities, it represents the primary tool for communicating with others. For example, people suffering from deafness and mutism convey messages predominantly through hand gestures, formally known as sign language. Apart from that, hand gesture can also be extremely effective at commanding. For example, hand gestures can be utilized in human computer interaction (HCI) to allow users to command a machine to perform certain actions, and thus enable seamless interaction between a machine and its users.

Hand gesture is divisible into two types, static and dynamic hand gesture. Static hand gesture is in the form of an image, while dynamic hand gesture is in the form of a series of frames, i.e. a video. This study focuses on static hand gesture recognition. Recognizing static hand gestures is challenging since a hand gesture image captured in a natural environment contains cluttered, and noisy background. The foremost challenge in recognizing hand gesture in an image is the background noise, while other prominent challenges include variation in illumination, viewpoint, size of the hand and skin color.

Manuscript received March 22, 2021; revised August 18, 2021. This work was supported by Fundamental Research Grant Scheme of the Ministry of Higher Education under award number FRGS/1/2019/ICT02/MMU/03/7.

Y.S. Tan is a student in the Faculty of Information Science and Technology, Multimedia University, Melaka, 75450 Malaysia. (e-mail: tanygsn@gmail.com)

K.M. Lim, the corresponding author, is a Lecturer in the Faculty of Information Science and Technology, Multimedia University, Melaka, 75450 Malaysia. (phone: 606-2523066; e-mail: kmlim@mmu.edu.my)

C.P. Lee is a Senior Lecturer in the Faculty of Information Science and Technology, Multimedia University, Melaka, 75450 Malaysia. (e-mail: cplee@mmu.edu.my)

Needless to say, while it has its challenges, developing a hand gesture recognition system can be worthwhile since it can entail many applications which are valuable and useful. In this paper, we leverage Wide Residual Network (WRN) for hand gesture recognition. The operation of a Convolutional (Conv) layer in residual block of WRN is sequenced as follow, batch normalization, Rectified Linear Unit and convolution, otherwise known as pre-activation residual block. The batch normalization alleviates the internal covariate shift and expedites the training process by normalizing the input distribution of the mini batches.

The main contributions of this paper are:

- We adopted a Wide Residual Network with pre-activation residual block for hand gesture recognition.
- We performed data augmentation techniques to increase the size of training data and to enhance the generalization capability of Wide Residual Network.
- We evaluated the performance of the Wide Residual Network and Convolutional Neural Network variants on three hand gesture datasets.

II. Related Works

In early work, Pugeault and Bowden [1] constructed an American Sign Language (ASL) dataset, and obtained 75% accuracy for 24 classes of alphabets in ASL by using Gabor filters for extracting features and Random Forest (RF) as the classifier. In other work [2], RF was utilized as a classifier for classifying features extracted by random line segments. Later on, Liu, Fan and Zhongzhang [3] utilized orientation segmentation feature for extracting features, and RF is used to classify the features. They achieved 94.33% accuracy for hand signed digits recognition. In a later work [4], RF was used to classify features extracted by distance adaptive scheme along with joint angles with constraints. They obtained 91.85% accuracy on ASL dataset with 24 gestures.

Principal component analysis (PCA) was utilized as the feature extractor in [5], and the features are classified by $k-$Nearest Neighbour ($k-$NN). They reported 92.75% accuracy for 20 gestures in ASL. Later on, $k-$NN was used to classify features of PCA in [6], features of Histogram of Oriented Gradients (HOG) and Scale Invariant Feature Transform (SIFT) in [7]. The former recorded an accuracy of 60.6% for 10 classes of ASL, while the latter achieved 90% accuracy for 20 gestures of Indian Sign Language.

In another work [8], Sparse Auto-Encoder (SAE) was used in conjunction with PCA to extract features from color and depth images. Classification of the extracted features was performed by Support Vector Machine (SVM). They were able to obtain 99.05% accuracy for 24 gestures in the ASL dataset. In [9], SVM was used to classify features of Local Binary Patterns histogram from color and depth images, and

geometric features of the hand based on skeletal information. An accuracy of 92.14% was obtained for the ASL dataset with 24 gestures. Later on, SVM was utilized to classify features extracted by Histogram of 3D facets (H3DF) in [10], and temporal pyramid matching of local binary subpatterns in [11].

Convolutional Neural Network (CNN) was introduced in [12] for image classification tasks, and it has been adopted by many researchers for hand gesture recognition as well [13, 14, 15, 16, 17, 18, 19]. In [20], 98.5% and 73.4% accuracies were obtained for NTU-HD dataset and HUST-ASL dataset, respectively. In another work [21], 91.26% accuracy was obtained for Thai fingerspelling dataset. On the other hand, CNN was also utilized to classify Japanese Sign Language in [22] and [23], where 96.1% accuracy was achieved for the former, and 93% accuracy was acquired for the latter. Additionally, variants of CNN were proposed in [24] and [25] for classifying ASL, both achieved 93.3% accuracy.

Several preprocessing steps were proposed in [26] for image enhancement before feeding the image into CNN for training and classification. In their work, they acquired 96.2% accuracy for Sign Language of Peru (LSP). Adapted Convolutional Neural Network (ADCNN) was proposed in [27] for classifying the alphabet of LSP. They obtained 84.5% accuracy. In another work [28], Directed Acyclic Graph (DAG) based CNN (DAG-CNN) was proposed for controlling robotic arms, they attained 84.5% accuracy on self-constructed dataset.

Although some previous works were able to achieve good results, they were only tested on a single dataset, and in some cases, a self-constructed dataset. Ergo, the robustness of the methods proposed in previous works was not put to the test. Hence, this study proposed Wide Residual Network (WRN) for static hand gesture recognition. In addition, in order to exhibit the robustness of the WRN, it is tested on three public datasets, where each dataset contains its own varied challenges.

III. Wide Residual Network (WRN)

This section details the proposed Wide Residual Network (WRN). Deep neural networks with hundreds or thousands of convolution layers stacked together become increasingly complex to train due to gradient vanishing problems. Gradient vanishing problem hinders the learning ability of the network as the gradients become increasingly small during back propagation, which severely impedes earlier layers ability to learn. Therefore it affects the generalization ability of the network and degrades the performance as training progresses. Residual Network (ResNet) [29] was invented to unravel gradient vanishing problem by making use of the identity shortcut connection, known as residual block. Identity shortcut connection bypass multiple layers to propagate $\boldsymbol{X}_l$ to latter layer by computing

$$\boldsymbol{X}_{l+1} = F(\boldsymbol{X}_l) + \boldsymbol{X}_l \tag{1}$$

where $+$ denotes the summation, and $\boldsymbol{X}_l$ and $\boldsymbol{X}_{l+1}$ are the input and output of the layers, and $F(\boldsymbol{X}_l)$ represents the output of the convolution layer given input $\boldsymbol{X}_l$, which is also known as the residual mappings to be learned. Fig. 1 shows the building block of the residual block.

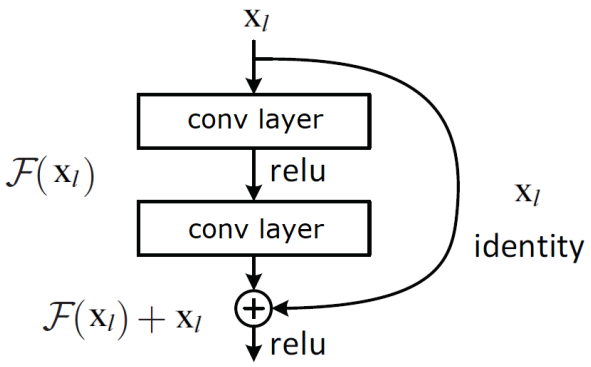

Fig. 1. Building Block of Residual Block

Despite the successful attempt at alleviating the gradient vanishing problem, it still suffers from the diminishing feature reuse problem. With ResNet, a small portion of improved accuracy demands roughly doubling the number of layers, while also entails long training hours. In view of this, Wide ResNet [30] was introduced to tackle these problems, primarily by increasing the width of the residual block, as the name "Wide ResNet" implies. Fig. 2 shows the difference in residual block between ResNet and Wide ResNet (WRN), operations such as batch normalization (BN) [31] and rectified linear unit (ReLU) activation [32] are omitted for clarity.

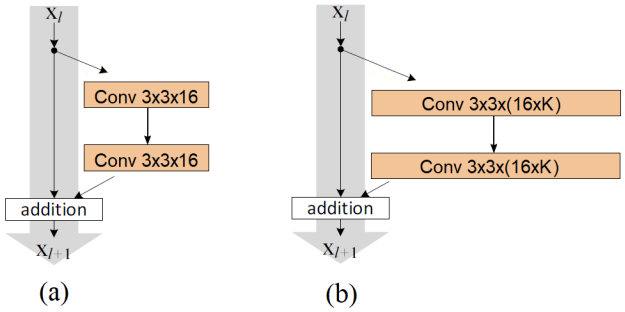

Fig. 2. (a) Residual Block in ResNet, (b) Wide Residual Block in WRN

Furthermore, rather than following conventional sequence of operations, which typically places BN and ReLu after convolution, WRN adopts the arrangement of placing BN and ReLu before convolution, which is known as pre-activation [33]. It has been demonstrated in the work that pre-activation is found to be the best performing placement for residual blocks. Fig. 3 shows the differences between conventional residual block and pre-activation residual block.

Additionally, dropout [34] is utilised in the wide residual block as well. In short, the proposed wide residual block contains the following operations in sequence: BN, ReLU, convolution, dropout, BN, ReLU, convolution, addition. Fig. 4 illustrates the proposed WRN architecture. 1×1 convolution layer is utilised before each group of wide residual blocks to match the number of feature maps required. Table I shows the architecture details of WRN.

IV. Experiments and Discussions

Three public datasets are utilized in our experiments. The details of the datasets are presented in this section,

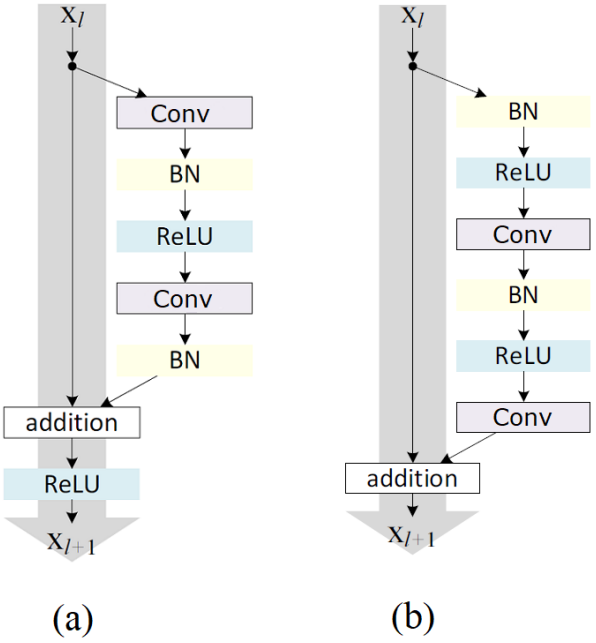

Fig. 3. The Sequence of Operations in (a) Conventional Residual Block, (b) Pre-activation Residual Block.

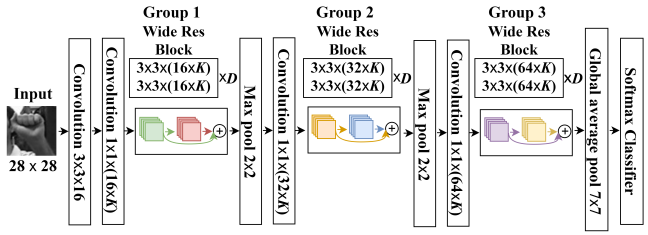

Fig. 4. The Proposed WRN Architecture, Which Consists of Three Groups of Wide Residual Blocks, Each Block with K as the Width of Convolution Layers, and Each Group with D as the Depth of Each Block in Succession.

TABLE I
ARCHITECTURE OF THE PROPOSED WRN

Block name	Output size	Operation type
convolutional layer	28×28	$\begin{bmatrix} 3 \times 3 \times 16 \end{bmatrix}$
Wide res block 1	28×28	$\begin{bmatrix} 3 \times 3 \times (16 \times K) \\ 3 \times 3 \times (16 \times K) \end{bmatrix} \times D$
Pooling layer	14×14	Max pool 2×2
Wide res block 2	14×14	$\begin{bmatrix} 3 \times 3 \times (32 \times K) \\ 3 \times 3 \times (32 \times K) \end{bmatrix} \times D$
Pooling layer	7×7	Max pool 2×2
Wide res block 3	7×7	$\begin{bmatrix} 3 \times 3 \times (64 \times K) \\ 3 \times 3 \times (64 \times K) \end{bmatrix} \times D$
Pooling layer	1×1	Global average pool 7×7

along with comprehensive evaluation of the proposed WRN, experimental results and results analysis.

A. Experimental Setup

ASL fingerspelling dataset [1] (termed as ASL dataset henceforth) comprises a total of 65,774 RGB images, which are converted into grayscale images, and subsequently resized into 28×28 pixels. It contains 24 gestures of ASL alphabets from A-Y (excluding letter 'J' and 'Z'). Images captured are signed by 5 signers with variation in viewpoint, background noise and illumination. Each gesture has approximately 500 images for each signer. Fig. 5 shows the sample images from the ASL dataset.

Fig. 5. Sample Hand Gesture Images from ASL Dataset.

ASL fingerspelling with digits [35] (termed as ASL with digits dataset henceforth) consists of 36 gestures in total. It contains 10 gestures for digits 0 to 9, and 26 gestures for alphabets including dynamic letters J and Z, in the form of static images. Gestures are captured with variation in viewpoint and illumination, in a black uniform background, and it is signed by 5 signers. This dataset consists of 2,515 RGB images in total, which are converted into grayscale and resized into 28×28 pixels. Fig. 6 shows the sample images of ASL with digits dataset.

Fig. 6. Sample Hand Gesture Images from ASL with Digits Dataset.

NUS hand gesture dataset [36] contains 2,000 RGB images of 10 gestures, all images are converted into grayscale and resized into 28×28 pixels. Each gesture has 200 images and images are captured with significant variation in background and illumination. Fig. 7 shows the sample images from NUS hand gesture dataset.

Fig. 7. Sample Hand Gesture Images from NUS Hand Gesture Dataset.

Due to the data-driven nature of CNN, it typically requires a huge amount of training data to generalize well on unseen data. In view of this, data augmentation is utilized to increase the number of images in the training set. Each image in the training set is manipulated by 9 data augmentation techniques, namely sigmoid correction, gamma correction, salt

and pepper, shearing, as well as perspective transformation techniques, which includes left or right skewing and corner skewing. In addition, the remaining 3 data augmentation techniques are combined manipulations, which are left or right skewing with salt and pepper, corner skewing with salt and pepper, shearing with salt and pepper. Fig. 8 illustrates the images generated by 9 aforementioned augmentation techniques along with the original image.

We adopted $k = 5$-fold cross validation for all of the experiments carried out in this paper. At each fold, $1/5$ of the images served as the testing set, while the remaining $4/5$ of the images were utilized as the training set. For the ASL dataset, each of the first four fold contains 52,619 images for the training set, and 13,155 images for the testing set, while the fifth fold has 52,620 images for the training set, 13,154 images for the testing set. By applying data augmentation to the training set, the training set with augmented data (with AD) contains 526,190 images for the first four fold, 526,200 images for the fifth fold. As for ASL with digits dataset, each fold contains 2,012 images for the training set, and 20,120 images for the training set with AD. The testing set for each fold contains 503 images. On the other hand, each fold in NUS hand gesture dataset comprises 1,600 images for the training set, 16,000 images for the training set with AD, and 400 images for the testing set.

B. Comprehensive Evaluation of WRN

In order to evaluate WRN performance comprehensively, we performed hyperparameters search for WRN. There are five hyperparameters in the proposed WRN, namely the depth of the wide residual block D, the width of the convolutional layer in a wide residual block K, dropout rate for convolutional layer in wide residual block P, as well as batch size B, and learning rate α.

By following the evaluation metric used in similar works on improving generalization of deep CNNs with data augmentation, as demonstrated in [37], [38], [39], [40], and [41]. The optimal value for each hyperparameter is determined based on the highest average recognition accuracy of all three datasets, either with or without augmented data (AD). A summary of the hyperparameters and the range of values tested are presented in Table II.

Table III illustrates the recognition accuracy at different D with and without augmented data. The depth of wide residual blocks, D, determines the number of blocks in succession within a wide residual block. Larger D entails greater computational cost while offering minor improvement in recognition accuracy. As illustrated in Table III, choosing D larger than optimal value not only requires higher computational cost, but also leads to diminishing return.

Apart from that, K determines the width of the convolutional layer in wide residual block. As depicted in Table IV, with and without augmented data, setting K larger than optimal value induces diminishing return while entailing greater computational cost, while smaller than optimal value leads to lower recognition accuracy.

Table V presents the recognition accuracy at different P with and without augmented data. This hyperparameter determines dropout probability for the convolutional layer in wide residual block. To put simply, it determines how much information in the feature maps are being retained and passed to the next convolutional layer. Choosing a larger P retains more information and likely triggers the network to overfit, while smaller P discards more information and possibly causes the network to underfit.

Table VI demonstrates the recognition accuracy at different B with and without augmented data. This hyperparameter determines how many images are fed to the network per iteration during training. Furthermore, Table VII shows the recognition rate at different α with and without augmented data. Specifically, α controls the amount of apportioned error that the weights of the network are updated with at each iteration during training. In other words, this hyperparameter controls the rate or speed at which the network learns.

It is noted that setting $D = 4$, $K = 3$, $P = 0.3$, $B = 16$ and $\alpha = 0.001$ produces the optimal average recognition accuracy on all three datasets with and without augmented data (AD).

TABLE II
SUMMARY OF HAND GESTURE RECOGNITION METHODS OPTIMAL HYPERPARAMETER SETTINGS

Hyperparameters	Tested Values	Optimal Value
Depth D	$2, 3, 4, 5$	4
Width K	$2, 3, 4$	3
Dropout probability P	$0.4, 0.3, 0.2, 0.1, 0$	0.3
Batch size B	$8, 16, 32$	16
Learning rate α	$0.01, 0.001, 0.0001$	0.001

TABLE III
RECOGNITION ACCURACY (%) AT DIFFERENT D WITH AND WITHOUT AUGMENTED DATA (AD)
$[K = 2, P = 0.2, B = 16, \alpha = 0.001]$

D		ASL	ASL with digits	NUS hand gesture	Average
2	Without AD	99.81	98.37	93.25	97.48
	With AD	99.99	99.24	97.70	98.98
3	Without AD	99.76	98.45	94.50	97.57
	With AD	99.99	99.20	97.60	98.93
4	Without AD	99.79	98.05	94.75	**97.53**
	With AD	99.99	99.05	98.05	**99.03**
5	Without AD	99.79	98.17	94.45	97.47
	With AD	99.99	98.97	97.85	98.94

TABLE IV
RECOGNITION ACCURACY (%) AT DIFFERENT K WITH AND WITHOUT AUGMENTED DATA (AD)
$[D = 4, P = 0.2, B = 16, \alpha = 0.001]$

K		ASL	ASL with digits	NUS hand gesture	Average
2	Without AD	99.79	98.05	94.75	97.53
	With AD	99.99	99.05	98.05	99.03
3	Without AD	99.82	98.25	95.10	**97.72**
	With AD	99.99	99.24	98.15	**99.13**
4	Without AD	99.79	98.05	94.55	97.46
	With AD	99.99	99.09	97.90	98.99

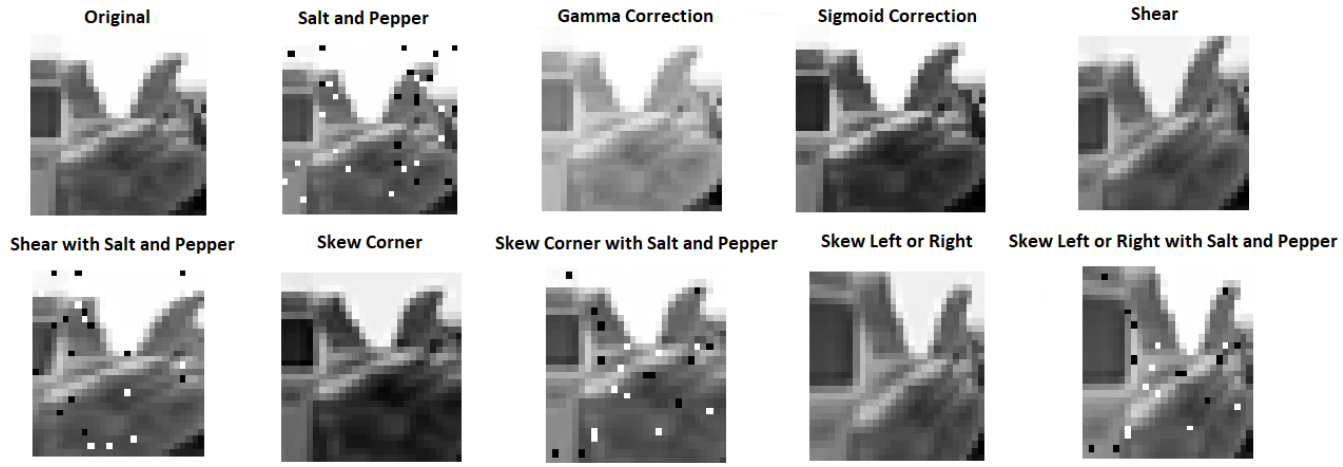

Fig. 8. A Hand Gesture Image and Its Augmented Images

TABLE V
RECOGNITION ACCURACY (%) AT DIFFERENT P WITH AND WITHOUT AUGMENTED DATA (AD)
$[D = 4, K = 3, B = 16, \alpha = 0.001]$

P		ASL	ASL with digits	NUS hand gesture	Average
0.4	Without AD	99.47	98.13	93.60	97.07
	With AD	99.97	99.13	98.00	99.03
0.3	Without AD	99.72	98.29	93.20	**97.07**
	With AD	99.99	99.28	98.25	**99.17**
0.2	Without AD	99.82	98.25	95.10	97.72
	With AD	99.99	99.24	98.15	99.13
0.1	Without AD	99.84	98.41	94.40	97.55
	With AD	99.99	99.24	98.00	99.08
0	Without AD	99.86	98.33	93.05	97.08
	With AD	99.99	99.20	97.60	98.93

TABLE VI
RECOGNITION ACCURACY (%) AT DIFFERENT B WITH AND WITHOUT AUGMENTED DATA (AD)
$[D = 4, K = 3, P = 0.3, \alpha = 0.001]$

B		ASL	ASL with digits	NUS hand gesture	Average
8	Without AD	99.77	98.17	94.00	97.31
	With AD	99.98	98.97	98.35	99.10
16	Without AD	99.72	98.29	93.20	**97.07**
	With AD	99.99	99.28	98.25	**99.17**
32	Without AD	99.75	98.01	93.85	97.20
	With AD	99.96	99.24	97.95	99.05

C. Experimental Results and Discussions

The hyperparameter settings of WRN are as follows, $D = 4, K = 3$, dropout rate of 30% $P = 0.3$, batch size of $B = 16$, learning rate of $\alpha = 0.001$ with Adam optimizer. The performance of WRN on 5-fold cross validation on all three datasets, with and without augmented data are summarized in Table VIII. WRN obtained a test accuracy of 99.99% for ASL dataset with augmented data, 99.28% for ASL with digits dataset with augmented data, 98.25% for NUS hand gesture dataset.

Additionally, we compared the performance of WRN with three existing CNN variants proposed for vision-based static hand gesture recognition, which includes a variant of AlexNet, termed as CNN baseline [26], Adapted Deep Convolutional Neural Network (ADCNN) [27], and Direct Acyclic Graph-based Convolutional Neural Network (DAG-CNN) [28].

TABLE VII
RECOGNITION ACCURACY (%) AT DIFFERENT α WITH AND WITHOUT AUGMENTED DATA (AD)
$[D = 4, K = 3, P = 0.3, B = 16]$

α		ASL	ASL with digits	NUS hand gesture	Average
0.01	Without AD	98.81	96.34	87.85	94.33
	With AD	99.91	98.73	94.65	97.76
0.001	Without AD	99.72	98.29	93.20	**97.07**
	With AD	99.99	99.28	98.25	**99.17**
0.0001	Without AD	99.73	97.42	84.30	93.82
	With AD	99.96	99.24	97.60	98.93

TABLE VIII
THE RESULTS OF 5−FOLD CROSS VALIDATION FOR WRN, WITH AND WITHOUT AUGMENTED DATA (AD)

Dataset	$k = 5$-Fold Cross Validation	Test accuracy (%) (without AD)	Test accuracy (%) (with AD)
ASL	1	99.76	99.98
	2	99.76	99.98
	3	99.59	100
	4	99.73	99.99
	5	99.77	99.98
	Average	99.72	99.99
ASL with digits	1	98.81	99.60
	2	98.01	99.40
	3	98.81	99.60
	4	98.21	98.61
	5	97.61	99.20
	Average	98.29	99.28
NUS hand gesture	1	91.50	98.25
	2	95.25	98.50
	3	93.25	98.00
	4	93.00	97.75
	5	93.00	98.75
	Average	93.20	98.25

For a fair comparison, batch normalization is applied to convolution layers, as well as fully-connected layers for all networks tested. In the meantime, softmax cross entropy is utilized as the loss function, and Adam optimizer for optimization, while other network architecture settings remain intact. Besides, batch size of 16 and learning rate of 0.001 are utilized for training on all three datasets. In addition, the number of epochs for training ASL dataset is set to 50, and 200 epochs for both ASL with digits and NUS hand gesture datasets. The performance comparison is summarized in Table IX.

TABLE IX
PERFORMANCE COMPARISON OF THE METHODS FOR STATIC HAND GESTURE RECOGNITION, IN TERMS OF RECOGNITION ACCURACY (%).

Method	Dataset	Accuracy % (Without AD)	Accuracy % (With AD)
CNN Baseline A [26]	ASL	99.85	99.95
	ASL with digits	98.69	99.32
	NUS hand gesture	89.15	94.35
	Average	95.90	97.87
ADCNN [27]	ASL	98.50	99.28
	ASL with Digits	98.49	98.97
	NUS Hand Gesture	83.10	89.30
	Average	93.36	95.85
DAG-CNN [28]	ASL	99.89	99.99
	ASL with digits	98.13	99.28
	NUS hand gesture	91.05	96.85
	Average	96.36	98.71
WRN (Proposed)	ASL	99.72	99.99
	ASL with digits	98.29	99.28
	NUS hand gesture	93.20	98.25
	Average	**97.07**	**99.17**

From the experimental results shown in Table IX, a consistent improvement in accuracy for all models with augmented data was observed. For ASL dataset, DAG-CNN achieves the highest accuracy without augmented data, while WRN obtains the highest accuracy with augmented data. As for ASL with digits dataset, CNN Baseline attains the highest accuracy with and without augmented data. However, for the NUS hand gesture dataset which is a significantly more challenging dataset, WRN achieves the highest accuracy, both with and without augmented data. In overall, WRN achieves the highest average recognition accuracy across all three datasets for both without and with augmented data.

D. Results Analysis

Confusion matrices for WRN on all three datasets are presented in Fig. 9, Fig. 10 and Fig. 11, respectively. In the ASL dataset, a total of 16 images are misclassified. The alphabets K and W are both misclassified three times each, M and U are misclassified 2 times each, alphabets A, B, G, N, O and R are all misclassified one time respectively.

For the ASL with digits dataset, a total of 18 images are misclassified. Alphabet O is misclassified five times to digit 0. This is due to both classes having the gestures with very little difference. The digit 1 is misclassified as alphabet Z three times. Both 1 and Z have the same gestures, the only distinction is the tilted gesture of the alphabet Z.

In the NUS hand gesture dataset, a total of forty images are misclassified. The gesture class 4 is misclassified the most, followed by gesture class 2, with eight times and six times respectively. Gesture class 4 is misclassified as gesture class 1 four times out of the total eight times. In addition, gesture class 1 is misclassified 5 times, followed by gesture class 8 with four images misclassified. The images misclassified by WRN on all three datasets are illustrated in Fig. 12.

V. CONCLUSION

The proposed WRN are experimented on three public datasets, and numerous data augmentation techniques are utilized to increase the size of the training set, in order to enhance the generalization. The results demonstrate consistent improvement in test accuracy across all three datasets when data augmentation is applied to images in the training set. WRN achieves a recognition rate of 99.99% for ASL dataset with augmented data, 99.28% for ASL with digits dataset with augmented data, and 98.25% for NUS hand gesture dataset with augmented data. Comparisons are made with existing CNN variants proposed for vision-based static hand gesture recognition. WRN outperforms all other deep learning models in comparison, and obtains the highest average test accuracy across all three datasets, which corroborates its robustness in vision-based hand gesture recognition.

REFERENCES

[1] N. Pugeault and R. Bowden, "Spelling it out: Real-time asl fingerspelling recognition," in *2011 IEEE International conference on computer vision workshops (ICCV workshops)*. IEEE, 2011, pp. 1114–1119.

[2] W. Nai, Y. Liu, D. Rempel, and Y. Wang, "Fast hand posture classification using depth features extracted from random line segments," *Pattern Recognition*, vol. 65, pp. 1–10, 2017.

[3] W. Liu, Y. Fan, T. Lei, and Z. Zhang, "Human gesture recognition using orientation segmentation feature on random rorest," in *2014 IEEE China Summit & International Conference on Signal and Information Processing (ChinaSIP)*. IEEE, 2014, pp. 480–484.

[4] C. Dong, M. C. Leu, and Z. Yin, "American sign language alphabet recognition using microsoft kinect," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2015, pp. 44–52.

[5] S. Upendran and A. Thamizharasi, "American sign language interpreter system for deaf and dumb individuals," in *2014 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)*. IEEE, 2014, pp. 1477–1481.

[6] D. Aryanie and Y. Heryadi, "American sign language-based finger-spelling recognition using k-nearest neighbors classifier," in *2015 3rd International Conference on Information and Communication Technology (ICoICT)*. IEEE, 2015, pp. 533–536.

[7] B. Gupta, P. Shukla, and A. Mittal, "K-nearest correlated neighbor classification for indian sign language gesture recognition using feature fusion," in *2016 International Conference on Computer Communication and Informatics (ICCCI)*. IEEE, 2016, pp. 1–5.

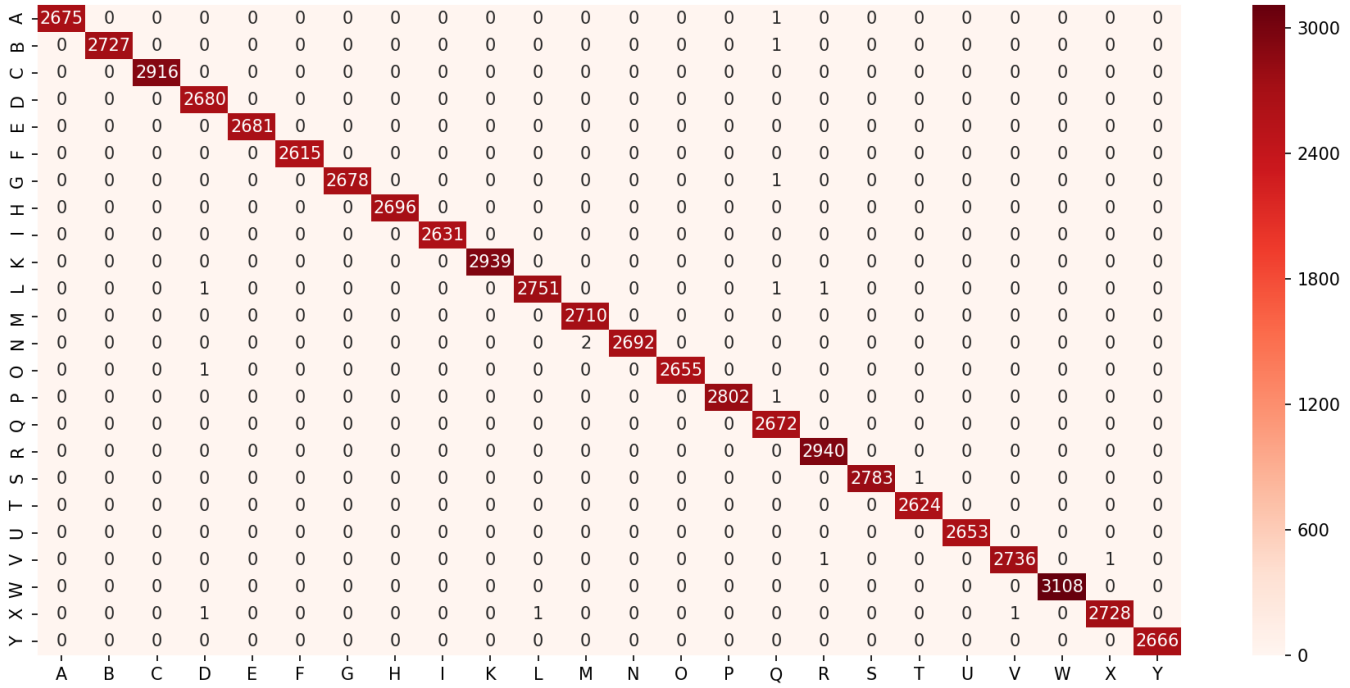

Fig. 9. Confusion Matrix for ASL Dataset (with Augmented Data)

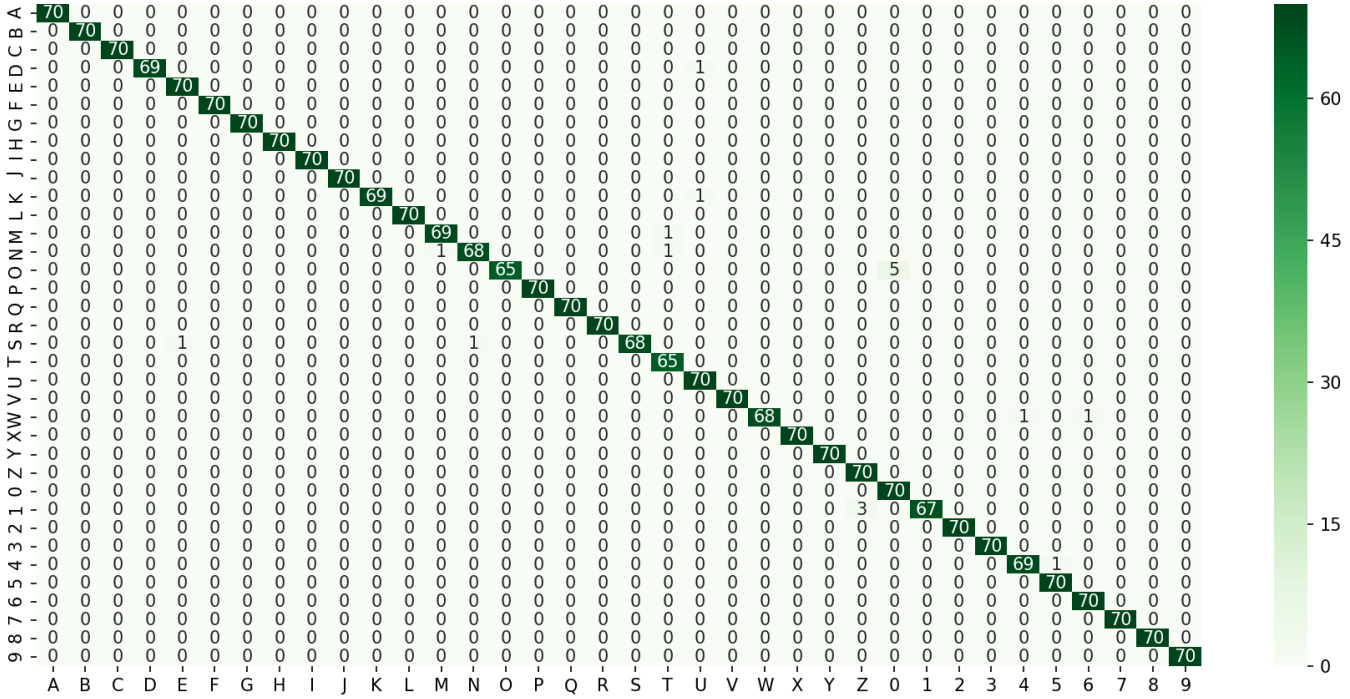

Fig. 10. Confusion Matrix for ASL with Digits Dataset (with Augmented Data)

[8] S.-Z. Li, B. Yu, W. Wu, S.-Z. Su, and R.-R. Ji, "Feature learning based on sae–pca network for human gesture recognition in rgbd images," *Neurocomputing*, vol. 151, pp. 565–573, 2015.

[9] C. S. Weerasekera, M. H. Jaward, and N. Kamrani, "Robust asl fingerspelling recognition using local binary patterns and geometric features," in *2013 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*. IEEE, 2013, pp. 1–8.

[10] C. Zhang and Y. Tian, "Histogram of 3d facets: A depth descriptor for human action and hand gesture recognition," *Computer Vision and Image Understanding*, vol. 139, pp. 29–39, 2015.

[11] A. I. Maqueda, C. R. del Blanco, F. Jaureguizar, and N. García, "Temporal pyramid matching of local binary subpatterns for hand-gesture recognition," *IEEE signal processing letters*, vol. 23, no. 8, pp. 1037–1041, 2016.

[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[13] K. M. Lim, A. W. Tan, and S. C. Tan, "A four dukkha state-space model for hand tracking," *Neurocomputing*, vol. 267, pp. 311–319, 2017.

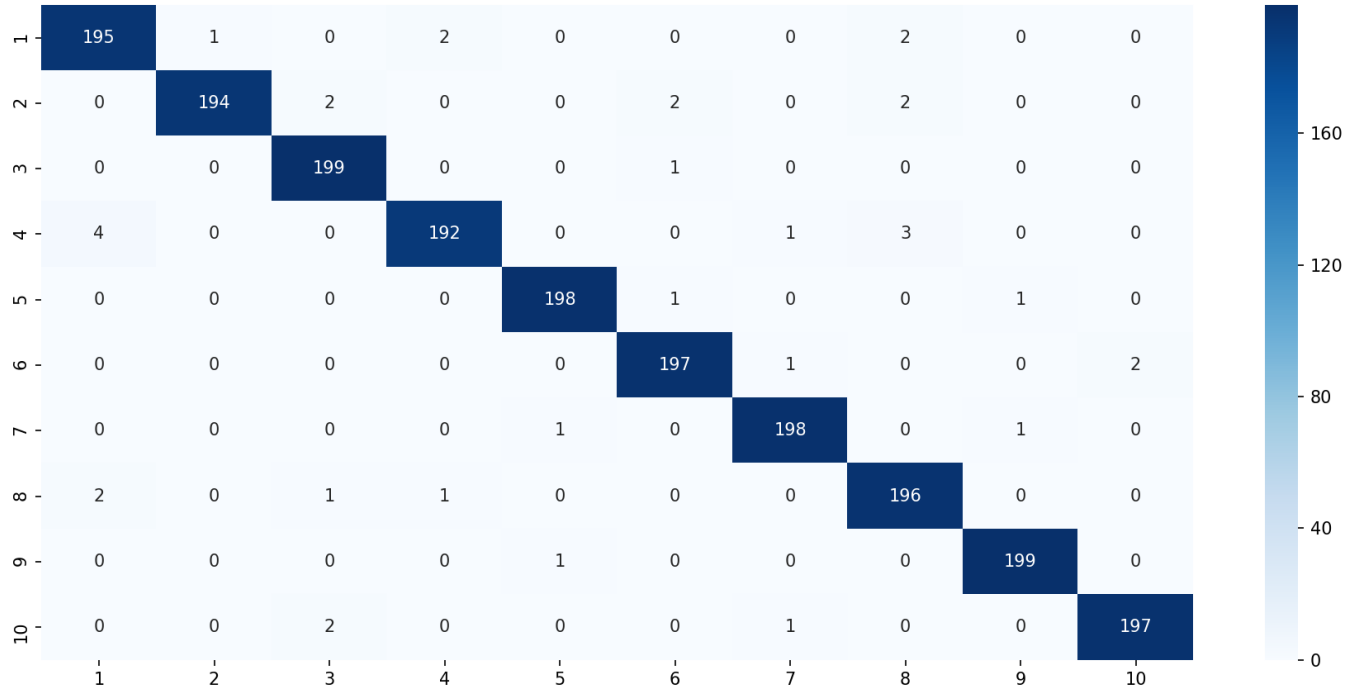

Fig. 11. Confusion Matrix for NUS Hand Gesture Dataset (with Augmented Data)

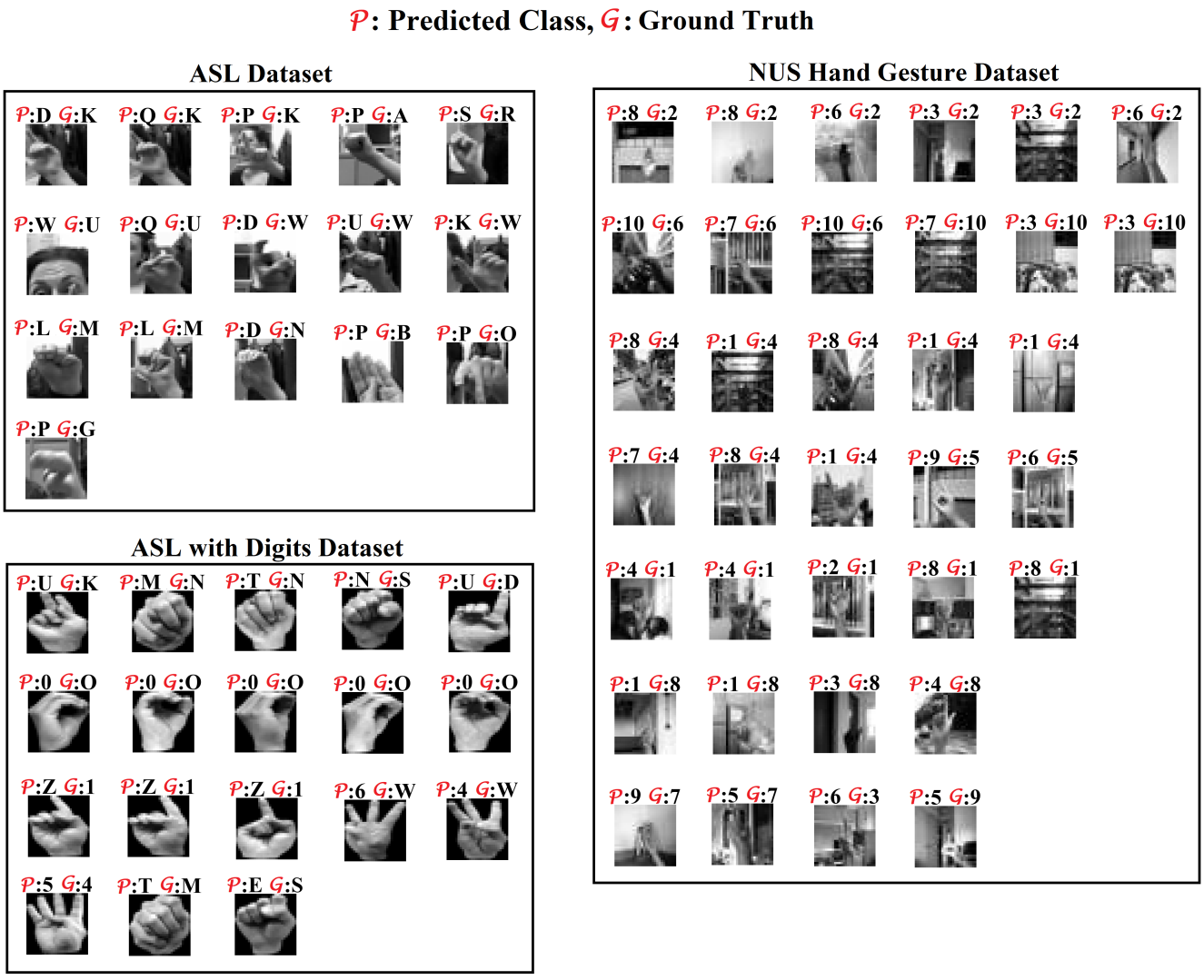

Fig. 12. Images Misclassified by WRN from ASL Dataset, ASL with Digits Dataset and NUS Hand Gesture Dataset

[14] Q. Zheng, X. Tian, S. Liu, M. Yang, H. Wang, and J. Yang, "Static hand gesture recognition based on gaussian mixture model and partial differential equation." *IAENG International Journal of Computer Science*, vol. 45, no. 4, pp. 569–583, 2018.

[15] J. Li, M. Yang, Y. Liu, Y. Wang, Q. Zheng, and D. Wang, "Dynamic hand gesture recognition using multi-direction 3d convolutional neural networks." *Engineering Letters*, vol. 27, no. 3, pp. 490–500, 2019.

[16] Y. Liu, M. Yang, J. Li, Q. Zheng, and D. Wang, "Dynamic hand gesture recognition using 2d convolutional neural network." *Engineering Letters*, vol. 28, no. 1, pp. 243–254, 2020.

[17] K. M. Lim, A. W. C. Tan, C. P. Lee, and S. C. Tan, "Isolated sign language recognition using convolutional neural network hand modelling and hand energy image," *Multimedia Tools and Applications*, vol. 78, no. 14, pp. 19 917–19 944, 2019.

[18] Y. S. Tan, K. M. Lim, C. Tee, C. P. Lee, and C. Y. Low, "Convolutional neural network with spatial pyramid pooling for hand gesture recognition," *Neural Computing and Applications*, pp. 1–13, 2020.

[19] Y. S. Tan, K. M. Lim, and C. P. Lee, "Hand gesture recognition via enhanced densely connected convolutional neural network," *Expert Systems with Applications*, p. 114797, 2021.

[20] Y. Li, X. Wang, W. Liu, and B. Feng, "Deep attention network for joint hand gesture localization and recognition using static rgb-d images," *Information Sciences*, vol. 441, pp. 66–78, 2018.

[21] P. Nakjai and T. Katanyukul, "Hand sign recognition for thai finger spelling: an application of convolution neural network," *Journal of Signal Processing Systems*, vol. 91, no. 2, pp. 131–146, 2019.

[22] B. Kwolek and S. Sako, "Learning siamese features for finger spelling recognition," in *International Conference on Advanced Concepts for Intelligent Vision Systems*. Springer, 2017, pp. 225–236.

[23] H. Hosoe, S. Sako, and B. Kwolek, "Recognition of jsl finger spelling using convolutional neural networks," in *2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA)*. IEEE, 2017, pp. 85–88.

[24] Q. Gao, J. Liu, Z. Ju, Y. Li, T. Zhang, and L. Zhang, "Static hand gesture recognition with parallel cnns for space human-robot interaction," in *International Conference on Intelligent Robotics and Applications*. Springer, 2017, pp. 462–473.

[25] K. Kania and U. Markowska-Kaczmar, "American sign language fingerspelling recognition using wide residual networks," in *International Conference on Artificial Intelligence and Soft Computing*. Springer, 2018, pp. 97–107.

[26] C. J. L. Flores, A. G. Cutipa, and R. L. Enciso, "Application of convolutional neural networks for static hand gestures recognition under different invariant features," in *2017 IEEE XXIV International Conference on Electronics, Electrical Engineering and Computing (INTERCON)*. IEEE, 2017, pp. 1–4.

[27] A. A. Alani, G. Cosma, A. Taherkhani, and T. McGinnity, "Hand gesture recognition using an adapted convolutional neural network with data augmentation," in *2018 4th International conference on information management (ICIM)*. IEEE, 2018, pp. 5–12.

[28] J. O. P. Arenas, R. J. Moreno, and R. D. H. Beleño, "Convolutional neural network with a dag architecture for control of a robotic arm by means of hand gestures," *Contemporary Engineering Sciences*, vol. 11, no. 12, pp. 547–557, 2018.

[29] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[30] S. Zagoruyko and N. Komodakis, "Wide residual networks," *arXiv preprint arXiv:1605.07146*, 2016.

[31] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[32] G. E. Dahl, T. N. Sainath, and G. E. Hinton, "Improving deep neural networks for lvcsr using rectified linear units and dropout," in *2013 IEEE international conference on acoustics, speech and signal processing*. IEEE, 2013, pp. 8609–8613.

[33] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *European conference on computer vision*. Springer, 2016, pp. 630–645.

[34] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[35] A. Barczak, N. Reyes, M. Abastillas, A. Piccio, and T. Susnjak, "A new 2d static hand gesture colour image dataset for asl gestures," 2011.

[36] P. P. Kumar, P. Vadakkepat, and A. P. Loh, "Hand posture and face recognition using a fuzzy-rough approach," *International Journal of Humanoid Robotics*, vol. 7, no. 03, pp. 331–356, 2010.

[37] R. Wu, S. Yan, Y. Shan, Q. Dang, and G. Sun, "Deep image: Scaling up image recognition," *arXiv preprint arXiv:1501.02876*, 2015.

[38] X. Yu, X. Wu, C. Luo, and P. Ren, "Deep learning in remote sensing scene classification: a data augmentation enhanced convolutional neural network framework," *GIScience & Remote Sensing*, vol. 54, no. 5, pp. 741–758, 2017.

[39] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," *arXiv preprint arXiv:1708.04896*, 2017.

[40] L. Taylor and G. Nitschke, "Improving deep learning using generic data augmentation," *arXiv preprint arXiv:1708.06020*, 2017.

[41] R. Takahashi, T. Matsubara, and K. Uehara, "Data augmentation using random image cropping and patching for deep cnns," *IEEE Transactions on Circuits and Systems for Video Technology*, 2019.