Development of CNN Transfer Learning for Dyslexia Handwriting Recognition

Conference Paper · August 2021

DOI: 10.1109/ICCSCE52189.2021.9530971

Iza sazanita Isa
Centre for Electrical Engineering StudiesUniversiti Teknologi MARA

Siti Azura Ramlan
Universiti Teknologi MARA (Pulau Pinang), Malaysia

Siti Noraini Sulaiman
MARA University of Technology

2021 11th IEEE International Conference on Control System, Computing and Engineering (ICCSCE) | 978-1-6654-1281-0/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICCSCE52189.2021.9530971

Development of CNN Transfer Learning for Dyslexia Handwriting Recognition

[1]Mohamed Syazwan Asyraf Bin Rosli, [2]Iza Sazanita Isa, Siti Azura Ramlan, Siti Noraini Sulaiman, Mohd Ikmal Fitri Maruzuki
School of Electrical Engineering, College of Engineering,
Universiti Teknologi MARA, Cawangan Pulau Pinang
Permatang Pauh, Pulau Pinang, Malaysia
[1]syazwanasyrafrosli@gmail.com, [2]izasazanita@uitm.edu.my

***Abstract*— Dyslexia is categorized as learning disorder that influence the ability of reading, writing and spelling. In Malaysia, "Instrumen Senarai Semak Disleksia (ISD)" that is provided by Ministry of Education is used to detect dyslexic student at early stage. However, such evaluations are time consuming, non-standardize and can lead to a biasing result since the evaluation is based on the teacher's experiences with the student. Hence, this research focus on the development of dyslexic handwriting recognition. The purpose of this research is to develop a transfer learning of Dyslexia handwriting recognition by using Convolutional Neural Network (CNN) based on famous architecture of handwriting recognition using of LeNet-5. Data augmentation and pre-processing was employed to a total of 138,500 handwriting image dataset before feeding it into network. The hyper-parameter of the model was tuned and analyzed to classify the 3 classes of dyslexic handwriting. The developed CNN model has successfully achieved a remarkable accuracy of 95.34% in classifying 3 classes of dyslexic handwriting. From the result, the objective in developing the CNN model for dyslexia handwriting recognition was successfully achieved.**

Keywords—dyslexia, handwriting, convolutional neural network, classification, hyper-parameter

I. Introduction

Developmental dyslexia are complicated topics for researcher to study especially on how to detect this disability at early stage because it relates to brain function. On average, the Ministry of Education Malaysia reported that there is one dyslexic case recognized in every 20 students[1]. Dyslexia is a learning disability that is originated in neurobiological. It exhibited through struggles with accurate word recognizing and also by lowly performance in reading and writing[2]. Detecting dyslexia in suspected child need to be detected as early as possible because with an early detection, the dyslexic child can be cured with an effective intervention program.

In Malaysia, a suspected child will be referred to either a child psychiatrist or a pediatrician who will give a preliminary diagnosis which then confirmed by the clinical psychologist through further testing[2]. Before going through that phase, suspected student during standard 2 will be tested using "Instrumen Senarai Semak Disleksia (ISD)" that is provided by Ministry of Education. Such evaluations are time consuming, non-standardize and can lead to a biasing result since the evaluation is based on the teacher's experiences with the student. Other than that, the ISD only evaluate on the letter reversal which is just one components of problems that students with dyslexia face in writing.

Suspected dyslexic children are always experienced primary difficulties in phonological awareness like phonemic awareness and manipulation, reading smoothness, single word reading, and also spelling[3]. Due to this, difficulties in reading comprehension will lead to problem in written expression. From this, the aim of this research is to develop a transfer learning of Dyslexia handwriting recognition by using Artificial Intelligence where Convolutional Neural Network (CNN) is used in classifying 3 types of dyslexic handwriting that was suggested by Susan Barton, the founder of Bright Solution for Dyslexia.

CNN is a Deep Learning algorithm which an input dataset such as an image can be train and categorized by assigning importance (learnable weights and biases) to various aspects in the image. From the paper [4], it reviews on various technique on detecting the handwriting and the accuracy of CNN is high in almost every cases. In this paper, a total of 138,500 datasets was collected to train and test the developed model. The CNN model was developed based on famous architecture of handwriting recognition using CNN which is LeNet-5 produced by LeCun et al [5]. There are two sections of analysis. First section is analysis through the LeNet-5 model with different hyper-parameter tuning. From the result, a new model is developed to classify the dyslexic handwriting class. The experimental analysis of the new model is the second section of the analysis part.

II. Literature Review

Year by year, researchers have come out with different approach in detecting dyslexic kids at an early stage by using different approach and method such as through brain behavior and through their handwriting. Brain activity is one of the prominent researches held by researcher in order to detect dyslexia since the problem is based from different functioning of the brain itself. From the previous study in [6], proposed a technique that used Electroencephalography (EEG) average FFT index that focused on writing disorder. The output shows that there are differences in hemispheric activation in brain from the test perform. On the other hand, interest in detecting dyslexia through machine learning has started to gain popularity among researcher. Researched by [7], proposed an Artificial Neural Network that classify potential cases of dyslexic kids based on questionnaire. The result obtained was stated as satisfactory and can be used for starting phase in evaluation dyslexic child before going to specialist.

Detecting dyslexic kids through handwriting has become one of the prominent approaches in detecting dyslexic children. One of the common problems for dyslexic child face in writing is letter reversal. Many of the letters that dyslexic learners reverse when reading can also contribute to the same letters that they reverse when writing[8]. From study [9] proposed a technique that will be used for automated diagnosis of dyslexia and for estimation of difficulty level as controlled by the handwriting ability screening questionnaire. This research used digitized tablet to acquire handwriting and subsequently employed a composite parameterization for quantifying its kinetic features and hidden complexities. Researched [10] on assistive technology for dyslexic using accelerometer-based hand writing Recognition and analog interactive voice response system (IVRS). The result produced approximately 90% accuracy by distinguishing dysgraphia products from capable products. However, none of these applications has attempted to detect dyslexic children through handwriting by using CNN. To our knowledge, this is the first to propose this kind of detection system.

III. Dataset Preparation

The input for the system is the handwriting images of three different classes which is the normal handwriting, reversal handwriting and corrected handwriting that was suggested by Susan Barton, the founder of Bright Solution for Dyslexia[11]. Dataset was collected from 3 sources where uppercase letter is from NIST Special Database 19 [12] while lowercase letter is from Kaggle Dataset [13] and some datasets for testing is obtained from dyslexia students at Seberang Jaya primary school.

A. Datasets Management

Every single image in these datasets was revised in order to separate the actual shape of alphabet with the corrected or wrong alphabet shape. The selection by categories is the process where the correct shape of alphabet is classified under normal handwriting while the wrong alphabet shape and corrected is classified under corrected handwriting. For the reversal group, the normal handwriting dataset was mirrored which is horizontally flip in order to produce reversal datasets. Some normal alphabets were not mirrored because it will produce the same shape after the process. A total of 78275 for normal class datasets was produced while for reversal is 52196 and for corrected is 8029.

B. Datasets Augmentation and Pre-processing

Pre-processing is a preparation process for the image dataset to be transform into common form before fed into classifiers as the original datasets are in different sizes, resolutions and shapes[14]. The process flow in data augmentation and pre-processing is shown in Fig. 1. From dataset management procedure before, total dataset in each class was imbalanced. Having an imbalanced dataset between class can lead bias to the prediction of more common class. One of the best methods to counter it, is by using data augmentation. Data augmentation is a technique that function to artificially expand the size of training dataset where it generates a new sample from original samples by applying certain manipulation towards a single images[15],[16]. For the augmentation and pre-processing of dataset, a software called XnConvert version 1.80 was used.

As shown in Table I, the augment is done by using technique of noise injection that involve a matrix of random values injection, usually from a Gaussian distribution where it can improve CNN model on learning more robust feature[17]. Two intensity noise value; 1.5 and 2.0 was applied to both class image dataset (reversal and corrected).

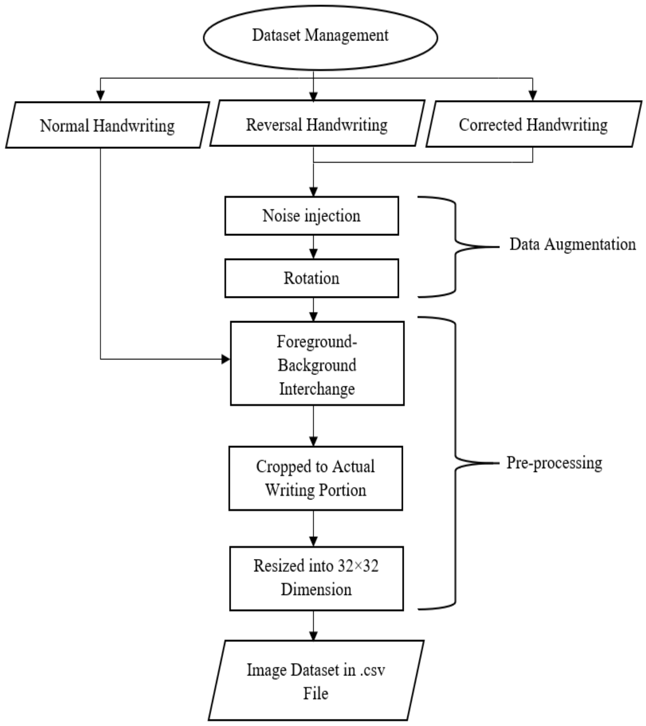


Fig. 1. Dataset augmentation and pre-processing

According to [17], the technique of rotation is implemented by rotating the image either to the left or to the right based on axis in between 1° and 359°. For rotation two angle is chosen which is 20º and -20º. From this process, dataset between classes was balanced where 78275 for normal, 77775 for reversal and 77304 for corrected. Next, for pre-processing, the foreground-background interchange was chosen in order to reduce computational overhead because having an image with more white point (value 1) than black point (value 0) will consume more power and memory on training the image[14]. This process will change the background to black while the handwriting in white. Lowercase 'd' was excluded from this foreground and background interchange because want to differentiate between reversal 'b' and normal 'd'. Next step is cropping the image to actual writing portion. This step will crop unwanted portion of image from bottom, top, right and left, hence the image will have focus to the alphabet at the centre. After that, the images were resized to 32×32 pixels, so that all datasets will have a uniform in size for the use as input of the CNN model. Finally, all the dataset was transformed into .csv file with label, where label 0 is Normal class, label 1 for Reversal class and label 2 is for Corrected class.

IV. CNN Model Development

For classification of three classes of dyslexic handwriting, a CNN approach is used. CNN was described as a leading architecture for most image identification, classification, and recognition duty[18]. In this project the CNN will work on the feature extraction, pattern analysis and classification of the handwriting image. CNN architectures have many variations but in general, CNN consist of convolutional and pooling layers, which is grouped into modules.

A. Software

To develop the CNN model, the software used is Jupyter under Anaconda package manager. For the programming language, Python version 3.7 is used.

B. Deep Learning Framework

Neural network framework is used to provide flexible APIs and configuration options for performance optimization where it is designed to facilitate and fasten the training of deep learning models. In this research, the neural network framework that is used is Keras with Tensorflow as backend. Tensorflow is low level while Keras is basically a high-level API.

C. LeNet-5 Model Hyper-parameter Tuning

In this research, the model was developed by adopting a famous CNN model for handwriting recognition which is LeNet-5 as depicted in Fig. 2. This model consists of 7 layers where the size of input image is 32×32 pixel. Having 3 convolutional layers (C1, C3 and C5), 2 sub-sampling (pooling) layers (S2 and S4), one fully connected layer (F6), and finally the output layer.

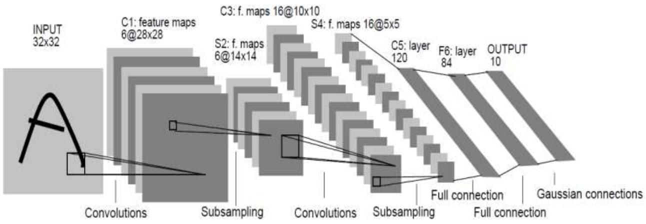


Fig. 2. Architecture of LeNet-5[5].

This model is used to train the 3 classes of dyslexic handwriting dataset. Some hyper-parameter tuning is done towards Lenet-5 model in order to analyze the suitable hyper-parameter for the given dataset. The aim in analyzing this, is to produce a modified model of LeNet-5 that efficient in classifying 3 classes of the input.

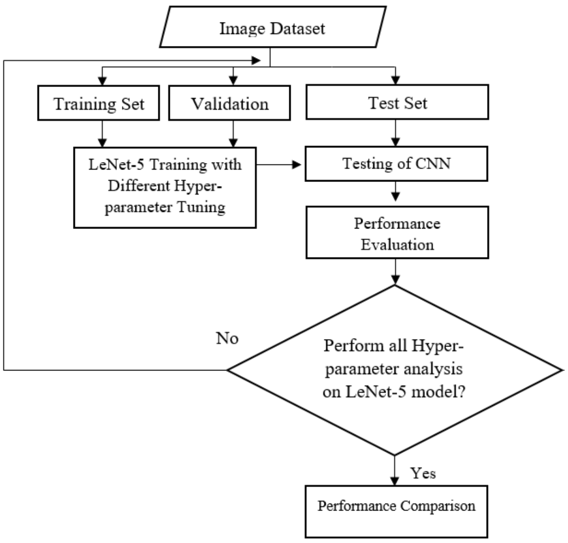


Fig. 3. Overall proposed work

Figure 3 shows the flowchart for analyzing the suitable hyper-parameter of LeNet-5. 70% of image dataset was allocated for training purpose, 20% for test and the remaining 10% was for validation purpose. Training dataset is used as sample to fit the model where the model sees and learns from this dataset. While tuning the model, the validation sample is used to perform an unbiased interpretation of the model fit on the training dataset. Validation sample is for evaluating the model frequently (once every epoch) as the model developed. Once the model finished training, the test sample was used to perform unbiased evaluation on the whole model fit. Next, the LeNet-5 model was trained and tested with different hyper-parameter such as varying the activation function (Tanh, ReLU, Sigmoid and Swish) and varying the optimizer (Adam, RMSprop and SGD). The CNN layer also was modified such as the type of pooling layer (average pooling and max pooling), adding batch normalization and dropout layer. All the result was analysed by evaluating the performance metrics such as training accuracy, training loss, validation accuracy, validation loss, test accuracy and test lost. Training time and test time also were evaluated.

D. Improved CNN Model Architecture

From the analysis made on LeNet-5 model, hyper-parameter that has better performance was chosen to combine and build a new improved model to classify the three classes of handwriting mistake. As shown in Fig. 4, the modified CNN network that has been changed are as follows.

1) *Type of pooling layer (Blue box in Fig. 4)*
   Pooling layer acts to modify the input feature into a representation of statistical results of surround feature, in order to produce smaller feature size than the previous one[19]. Both pooling layer change from using average pooling to max pooling.
2) *Batch normalization layer (Yellow box in Fig. 4)*
   Adding batch normalization layer after every convolutional layer. Batch normalization layer is a layer that normalize the hidden unit's activation values so the use of this activation remains the same for the training which it also helps to improve the accuracy and speed up the training process[20]
3) *Dropout Layer (Red box in Fig. 4)*
   This layer was added after the third convolutional layer. Dropout layer is a layer that apply a regulations technique that randomly selected neurons are ignored during training. The dropped neurons will not commit on forward pass and any weight updates are not used to the neuron on the backward pass[21]. Main function of dropout is to reduce overfitting of the model.
4) *Activation Function*
   This change was applied in every convolutional layer. Original model use Tanh but for this model, Swish function is used.
5) *Optimizer Algorithm*
   In this modified model, Adam optimizer is used. One of the configuration parameters for Adam is the learning rate with value of 0.001[22].

Convolutional layer work as feature extractor where it learns the feature representations of the input images which the trainable convolutional kernel will regulate its kernel weights automatically in backpropagation training process[4], [19]. After all feature extraction is done, the model needs to classify the data, this can be done using a fully connected layer. A softmax function is a multiclass logistic classifier and is used at fully connected layer before output since the model want to classify multiclass[19].

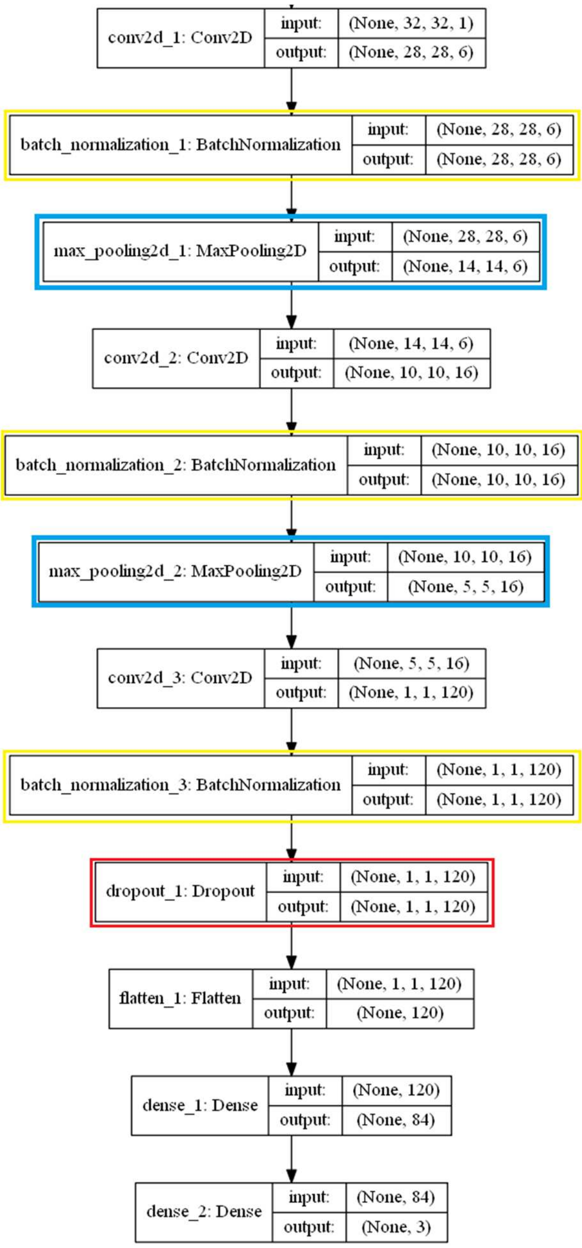


Fig. 4. Modified CNN layer.

V. Experimental Result And Discussion

In this section, the analysis of LeNet-5 hyper-parameter was done and how the results effect the development of new modified CNN model in order to classify 3 classes of dyslexic handwriting. Then, the new model also was analyzed on the metrics performance. Both LeNet-5 and modified model were using the same dataset that had been augmented and pre-processing as input.

A. LeNet-5 Model Analysis

First hyper-parameter tuning for LeNet-5 is by varying activation function. Activation function will calculates the bias and weighted sum of input and later it will decide on either the neuron can be fired or not[23]. Original LeNet-5 model used Hyperbolic Tangent function (Tanh) as activation function. Other activation used for comparison is Sigmoid, Rectified Linear Unit (ReLU) and the newest activation function, Swish function. The result in Table III shows that, Swish function outperform others and have slightly better performance than ReLU in test accuracy and this fact is supported by the study performed by Ramachandran et al., 2017[24].

TABLE I. Activation Performance Comparison

Activation	Test Accuracy	Test Loss	Training Time, s	Test Time, s
TanH	0.8911	0.2848	316	6
ReLU	0.9018	0.2602	189	4
Sigmoid	0.3629	1.0980	161	6
Swish	0.9039	0.2601	195	4

Swish provide a simplicity and improved accuracy as it does not face vanishing gradient problems but produce good information propagation during training[24]. Unlike Sigmoid function that have problem with vanishing gradient. The difference between ReLU and Swish is the time for the model to converge. It found that ReLU converge faster compared to Swish but by increasing in epoch, the Swish shows positive incrementing of learning process. The original activation for LeNet-5 model, Tanh function produce an acceptable accuracy, but the training time are almost double the time for other activation function.

Second analysis is comparison on optimization algorithm. Optimization provides a way to minimize the loss function in training deep learning model[25]. There are 4 different types of optimization algorithm were tested including Adaptive moment estimation (Adam), Root Mean Square propagation (RMSprop) and Adaptive Gradient (Adagrad). Adam optimizer shows highest accuracy in both training and testing and lowest in training and test loss. It found that the Adam algorithm converge faster compared to RMSprop even though they produce almost the same accuracy. Adam perform well because it was designed by combining the advantages of Adagrad that perform well with sparse gradient, with RMSprop that performed well in non-stationary and online setting[22].

TABLE II. Optimizer Performance Comparison

Optimizer	Test Accuracy	Test Loss	Training Time, s	Test Time, s
Adam	0.9291	0.2025	170	6
RMSprop	0.9244	0.2029	172	4
Sgd	0.8911	0.2848	316	6
Adagrad	0.9004	0.2563	165	5

Next, the analysis is based on modifying the LeNet-5 layer. First modification analysis is by applying different pooling method which is average pooling and max pooling. Back then LeNet-5 model used subsampling which now known as average pooling. Average pooling is a selection that take average value of all pixels in the batch while max pooling takes maximum pixel value. For the max pooling surpass average pooling in performance on training accuracy and training loss, for test accuracy, max pooling produces 0.9397 beating the average pooling with 0.8911. Either max or average pooling, the performance is depending on the dataset and the features, and also the classification problem[26]. So, for this model, max pooling suits well with the dataset and the classification class.

Batch normalization is another modification made to the model. The layer was located right after convolutional layer. Table V shows the performance classification of the variation of where the layer placed in the neural network model. Adding this layer was proven in improving accuracy and reduce lost but most importantly it reduces the training time compare to without any batch normalization layer.

TABLE III. BATCH NORMALIZATION PLACEMENT COMPARISON.

Batch Normalization Layer	Test Accuracy	Test Loss	Training Time, s	Test Time, s
No layer added	0.8993	0.2657	194	6
After 1st Convolutional layer	0.9052	0.2493	135	3
After 1st and 2nd Convolutional layer	0.9205	0.2146	141	4
After all Convolutional layer	0.9389	0.1641	139	3

The results show that, by placing batch normalization layer after every convolutional layer produce higher accuracy. It also contributes to faster convergence compare to without batch normalization layer. This is due to its effect in reducing the amount by what the hidden unit values shift around (covariance shift)[20].

B. *Modified Model Performance Analysis*

A new modified CNN model aspired by LeNet-5 was developed by taking the best hyper-parameter from section before that can perform well in classifying the three classes. To reduce overfitting of the new model, a dropout layer was added to the CNN[25]. As shown in Table IV, the dropout value that is suitable for the modified model is 0.1.

TABLE IV. DROPOUT VALUE COMPARISON ON PERFORMANCE

Dropout 'p' Value	Train Accuracy	Validation Accuracy	Train Loss	Validation Loss
0.1	0.9823	0.9702	0.0469	0.0866
0.3	0.9735	0.9649	0.0688	0.0974
0.5	0.9610	0.9541	0.1028	0.1267

The performance of original LeNet-5 and the modified CNN model was analyzed by using metrics performance which is train accuracy and loss, and validation accuracy and loss. Both were analyzed with 20 epochs and using supervised learning. In supervised learning, the data was trained using labeled dataset according to its respective class.

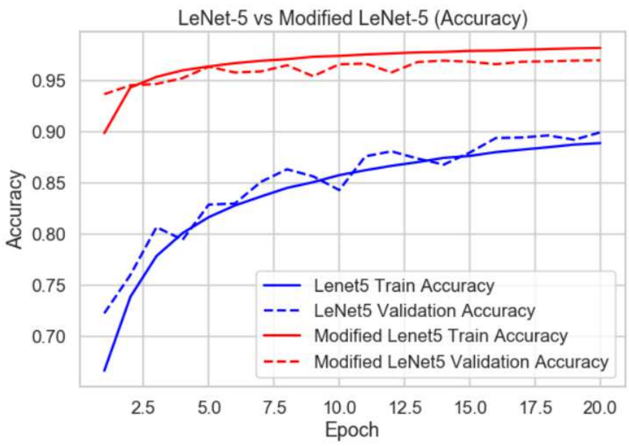


Fig. 5. Accuracy of LeNet-5 vs Modified LeNet-5 .

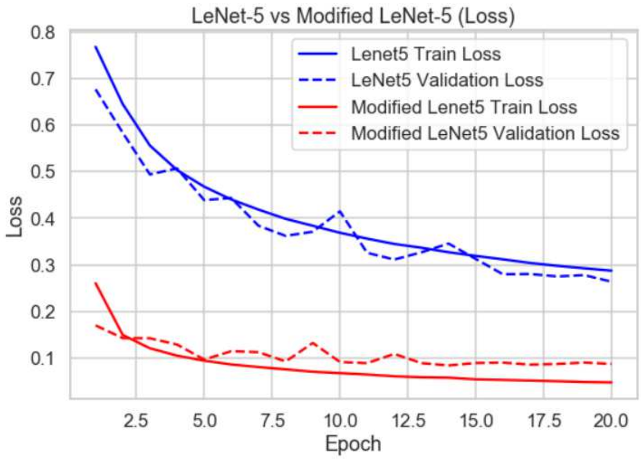


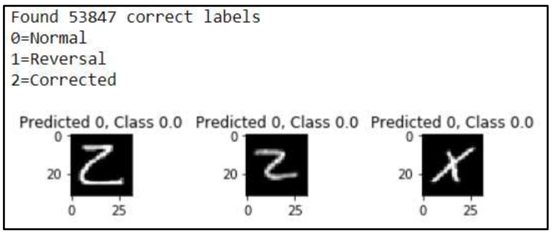
Fig. 6. Loss of LeNet-5 vs Modified LeNet-5 .

In Figure 6, the training accuracy and validation accuracy of modified LeNet-5 outperformed the original LeNet-5 model in classsifying three classes of dyslexic handwriting image. Modified LeNet-5 also converge faster at second epoch while the LeNet-5 model take more epoch to converge. The improvement of layer by adding batch normalization layer into modified LeNet-5 has proven in accelerating the learning process of the model. Comparing both validation accuracy in Figure 5 and validation loss in Figure 6, the original architecture shows a fluctuating graph while the modified model started to stable at 13th epoch. In this case, a regularisation is a need[21], [25]. The modified model uses a dropout as regularisation while the original model did not use any regularisation method. In addition, the incremental learning behaviour evident that the modified model shows its ability to learn and extract image features epoch by epoch. This ability is the benefits of CNN because it learns and does not depending on any specific images[19]. The optimizer also play an important role in handling a good model. The Adam optimizer help in tuning the modified model parameter such as weights[22] to helps in reducing the loss function.

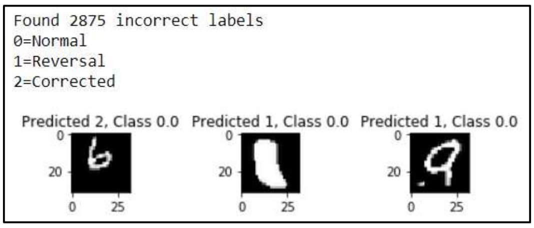
TABLE V. MODEL PERFORMANCE COMPARISON

Model	Test Accuracy	Test Loss	Training Time, s	Test Time, s
LeNet-5	0.8873	0.2887	272	5
Modified LeNet-5	0.9534	0.1583	329	4

Table V shows the performance comparison on both model on a new dataset which is test dataset. It shows that, the modified model produces higher accuracy and low loss compared to original LeNet-5 model. The training time is shorter for the LeNet-5 compared to modified model due to the modified model has more layer to execute such as batch normalization layer and dropout layer. Figure 7 shows example of image dataset that is correctly predict and incorrectly predict respectively for test dataset. The accuracy in classifying every class is shown in Table VI. Accuracy for normal class is lower compared to other class. The reason behind the lower accuracy in normal class is because of the data augmentation process. This process only involves reversal and corrected dataset due to the imbalanced dataset. This is proven that, augmented the dataset can lead to a better classification performance especially increase in accuracy[14], [15].



(a)



(b)

Fig. 7. (a) Correct prediction and (b) Incorrect prediction for test set.

TABLE VI. CLASSIFICATION PERFORMANCE OF PROPOSED MODEL

Class	Accuracy
Normal	0.92
Reversal	0.97
Corrected	0.97

VI. CONCLUSION

The modified CNN model that was inspired from famous LeNet-5 architecture was able to produce remarkable performance especially in accuracy of classifying 3 classes of dyslexic handwriting image. In achieving the high accuracy, some hyper-parameter tuning was performed in finding the best parameter to suit the modified model. In addition, dataset pre-processing and augmentation also helps in accelerating the classification accuracy. From the result, the objective in developing the CNN model for dyslexia handwriting recognition was successfully achieved. From this study, a number of future works can be done to improve the dyslexia handwriting recognition such as collecting more dyslexic handwriting image for test set in examining the performance of the model with fully real environment.

ACKNOWLEDGMENT

The authors have no conflict of interest and acknowledge the School of Electrical Engineering, College of Engineering and Research Management Unit, Universiti Teknologi MARA, Cawangan Pulau Pinang for providing financial support funded by UiTM grant with reference number: 600-TNCPI 5/3/DDN (07) (009/2020).

REFERENCES

[1] Komala, "Disleksia – Ramai Kanak kanak tidak kenal huruf," *Utusan Malaysia*, PD, p. 17, 22-Sep-2004.

[2] N. A. M. Yuzaidey, N. C. Din, M. Ahmad, N. Ibrahim, R. A. Razak, and D. Harun, "Interventions for children with dyslexia: A review on current intervention methods," *Medical Journal of Malaysia*, vol. 73, no. 5, pp. 311–320, 2018.

[3] Texas Education Code, "I . Definitions and Characteristics of Dyslexia," pp. 8–12.

[4] S. Shofia Rosyda and T. W. Purboyo, "A Review of Various Handwriting Recognition Methods," *International Journal of Applied Engineering Research ISSN*, vol. 13, no. 2, pp. 973–4562, 2018.

[5] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-Based Learning Applied to Documents Recognition," *proc. OF THE IEEE*, 1998.

[6] S. N. Farihah, K. Y. Lee, W. Mansor, N. B. Mohamad, Z. Mahmoodin, and S. A. Saidi, "EEG average FFT index for dyslexic children with writing disorder," *ISSBES 2015 - IEEE Student Symposium in Biomedical Engineering and Sciences: By the Student for the Student*, pp. 118–121, 2016.

[7] M. Kohli and T. V. Prasad, "Identifying dyslexic students by using artificial neural networks," *WCE 2010 - World Congress on Engineering 2010*, vol. 1, no. June 2010, pp. 118–121, 2010.

[8] J. E. Richmond and M. Taylor, "Visual recognition difficulties: Identifying primary school learners' directional confusion in writing letters and numbers," *South African Journal of Occupational Therapy*, vol. 44, no. 3, pp. 2–6, 2014.

[9] J. Mekyska, M. Faundez-Zanuy, Z. Mzourek, Z. Galaz, Z. Smekal, and S. Rosenblum, "Identification and Rating of Developmental Dysgraphia by Handwriting Analysis," *IEEE Transactions on Human-Machine Systems*, vol. 47, no. 2, pp. 235–248, 2017.

[10] G. Srivastav and A. Agarwal, "Assistive Technology for Dyslexic using Accelerometer based Hand Writing Recognition and Analog IVRS," pp. 17–22, 1994.

[11] S. Barton, "How to Spot Dyslexia in a Writing Sample by Susan Barton," 2017. [Online]. Available: https://vimeo.com/203740803. [Accessed: 23-Apr-2019].

[12] P. J. Grother, "NIST Special Database 19," *NIST*, 2016. [Online]. Available: https://www.nist.gov/srd/nist-special-database-19. [Accessed: 22-May-2019].

[13] S. Patel, "A-Z Handwritten Alphabets in .csv format," *Kaggle*, 2017. [Online]. Available: https://www.kaggle.com/sachinpatel21/az-handwritten-alphabets-in-csv-format. [Accessed: 22-May-2019].

[14] M. A. H. Akhand, M. Ahmed, and M. M. H. Rahman, "Convolutional Neural Network Training incorporating Rotation-Based Generated Patterns and Handwritten Numeral Recognition of Major Indian Convolutional Neural Network Training incorporating Rotation based Generated Patterns and Handwritten Numeral Recogni," no. July, 2017.

[15] Y. Shima, "Pattern Augmentation for Handwritten Digit Classification based on Combination of Pre-trained CNN and SVM," 2017.

[16] S. Tabik and D. Peralta, "A snapshot of image pre-processing for convolutional neural networks : case study of MNIST," vol. 10, pp. 555–568, 2017.

[17] C. Shorten and T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," *Journal of Big Data*, 2019.

[18] Z. W. Waseem Rawat, "Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review," *Neural Computation*, vol. 29, no. 9, pp. 2352–2449, 2017.

[19] M. Y. W. Teow, "Understanding convolutional neural networks using a minimal model for handwritten digit recognition," *Proceedings - 2017 IEEE 2nd International Conference on Automatic Control and Intelligent Systems, I2CACIS 2017*, vol. 2017-Decem, no. October, pp. 167–172, 2017.

[20] J. Bjorck, C. Gomes, B. Selman, and K. Q. Weinberger, "Understanding batch normalization," *Advances in Neural Information Processing Systems*, vol. 2018-Decem, no. NeurIPS, pp. 7694–7705, 2018.

[21] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," pp. 1–18, 2012.

[22] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," pp. 1–15, 2014.

[23] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, "Activation Functions: Comparison of trends in Practice and Research for Deep Learning," pp. 1–20, 2018.

[24] P. Ramachandran, B. Zoph, and Q. V. Le, "Searching for Activation Functions," pp. 1–13, 2017.

[25] S. V. G. Reddy, K. Thammi Reddy, and V. Valli Kumari, "Optimization of deep learning using various optimizers, loss functions and dropout," *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, no. 2S, pp. 272–279, 2018.

[26] Y. L. Boureau, J. Ponce, and Y. Lecun, "A theoretical analysis of feature pooling in visual recognition," *ICML 2010 - Proceedings, 27th International Conference on Machine Learning*, no. June 2014, pp. 111–118, 2010.