Convolutional Neural Network Based Image Classification And New Class Detection

Akshaya B
Department of Computer Science and Engineering
Government Engineering College
Thrissur, India
akshayabalaram@gmail.com

Prof. Kala M T
Department of Computer Science and Engineering
Government Engineering College
Thrissur, India
kalakrishnakumar22@gmail.com

2020 International Conference on Power, Instrumentation, Control and Computing (PICC) | 978-1-7281-7590-4/20/$31.00 ©2020 IEEE | DOI: 10.1109/PICC51425.2020.9362375

***Abstract*—Image Classification is the task of assigning an input image to a label from a set of fixed labels. This is one of the main problems in computer vision that have many practical applications. For any classification problem, the main aim is to achieve better classification accuracy. If the classification accuracy is less, then misclassification happens and this will leads to different kinds of problems. Many of the classification models only consider the existing class instances. When a new class instance arrives the classification model not detect it properly. They actually misclassified the new class instance into an existing class instance. The proposed method therefore shows a better accurate classification and new class detection model for images. Also if needed, then the new class can be added with the model to classify correctly in the future. Recent studies show that Convolutional Neural Network(CNN) can be effectively used for image classification tasks. So here creating this better accurate classification and new class detection model based on CNN. The detection of a new class is done by looking into the trend of the softmax prediction score of class labels. In this work, the model is built for CIFAR10 image dataset. This dataset is actually a complex dataset, so creating a model for this dataset can consider as a base and extended for the classification and new class detection in other images in different applications.**

***Index Terms*—Convolutional Neural Network, Image Classification, New Class Detection, Accuracy, Prediction**

I. Introduction

Classification is a technique that helps us to make decisions in our day to day life. There are many situations that arrive in life which require placing an object in a particular group or class based on the characteristics of that object, in such cases classification is necessary. Most of the problems in today's world are classification problems. Better classification accuracy always leads to accurate classification results. So that researchers try to find different classification methods to improve the classification accuracy. In today's digital world, people are sharing a large number of images through social media networks, apps, and websites. Also creating many images and videos using their mobile phones with the camera [1]. Production of numerous images in a single day makes the need for classification of images. This image classification will provide a proper arrangement of images that makes a fast and easy accessibility of images. Based on the characteristics of images classify it into different groups.

Classification problems are solved mathematically in a non-linear fashion, for different areas like forecasting weather, speech recognition, stock market prediction, etc. Image classification is very useful for different applications like automatic categorization of images in social media and search engines, image tagging, etc. Also for the efficient and better analysis of surrounding scenes, categorization will be helpful. The classification of indoor and outdoor scenes are sometimes difficult. Because it may contain blurry and noisy data. If the image faces problems due to noise and poor quality, in such case identification of the object in that image becomes very hard. Also if the image contains more objects then difficulty level again increases. This is why researchers create new classification methods for getting more accurate classification results. It is very important to choose the appropriate classification method for a classification problem.

The main objective in image classification is the identification of accurate features existing in the image. Because of the vast usage of image classification in different applications [2], there is always a need for finding a better accurate method to provide higher accurate classification results. Another main problem is that many of the previous works only consider existing class images. When time evolves there can be the arrival of new class images also. The classification model not detecting these new class images. Actually what happens is the classification model misclassifies the new class image into an existing class image. If there is a need for classifying this new class image correctly in the future then, need to upgrade the classification model for the new class also. For this a proper new class detection is required with the classification model.

In this paper, try to create a classification model that provides a good classification of images with approximately 90% classification accuracy. Classification accuracy is actually the ratio of correctly predicted results to the total number of predictions that occur with the model. Lesser classification accuracy can make more misclassified results during the prediction of test input images. So creating a better classification model for images. Also this classification model can detect new class images. So if needed, this classification model can upgrade to correctly classify new class images in the future. Convolutional Neural Network(CNN) is a class of deep neural network which has a layered structure. Because of that, they can be analyze images at various levels of abstraction [3]. Here building this model using CNN. A CNN

Authorized licensed use limited to: San Francisco State Univ. Downloaded on June 21,2021 at 18:20:17 UTC from IEEE Xplore. Restrictions apply.

architecture is designed and proposed for this work. Here considering CIFAR10 image dataset. CIFAR10 dataset is a complex dataset, because it contains 32x32 pixel tiny color images. So extracting features from these tiny images is very difficult. Because of that many of the previous CIFAR10 classification methods provide poor classification accuracy. So here building this better accurate classification and new class detection model for CIFAR10 image dataset. And this work can consider as a base and extend to build better accurate classification and new class detection model for other images in different applications.

Briefly discusses some of the previous classification works on CIFAR10 dataset here. Mark D. McDonnell et al. proposed a method (FLSCNN) which has randomly valued classifier stage input weights [4]. And then least-squares regression is used to train the classifier output weights in a single batch. It has linear classifier stage output units. The main problems with this work are, it suffering from reduced performance and dropout in the convolutional stage or data augmentation is required. Tsung-Han Chan et al. proposes another method (PCANet) in which network process images by cascaded principal component analysis(PCA) [5]. And it is followed by simple binary hashing and block-wise histograms for indexing and pooling. Drawbacks with this work are, not sufficient to handle variability and require normalization for good performance. Alexey Dosovitskiy et al. presented a method (DUFLCNN) in which CNN is trained using unlabeled data [6]. A feature learning algorithm is proposed here. Which is used to extract features from the images. The main problems with this work are, this is not adaptable to arbitrarily large dataset and it is computationally expensive. Alec Radford et al. proposed another method in that a DCGAN architecture is introduced [7]. This architecture is used to learn representation of images for supervised learning and generative modeling. Drawbacks with this works are, the model has instability and sometimes a subset of filters collapse to a single oscillating mode.

II. PROPOSED METHOD

There are many previous methods that perform image classification on CIFAR10 dataset. But most of them not achieve good classification accuracy. They have classification accuracy below 90%. So here, trying to make a classification model for CIFAR10 image classification with approximately 90% accuracy using CNN. Also new class detection and its updation with the classification model to correctly classify it in the future is done here. Dataset, proposed CNN architecture and implementation of this work are discussed in this section.

A. Dataset

In this work, using CIFAR10 image dataset which can be downloaded from the official website. It contains 32x32 tiny color images. This dataset has 60000 sample images, from which 50000 images are used for training and 10000 images are used for testing. It contains 10 different classes, each class contains 6000 images. The class labels in the dataset are airplane, automobile, bird, cat, deer, dog, frog, horse, ship, and truck. Some sample images in the CIFAR10 dataset are shown in the Fig. 1.

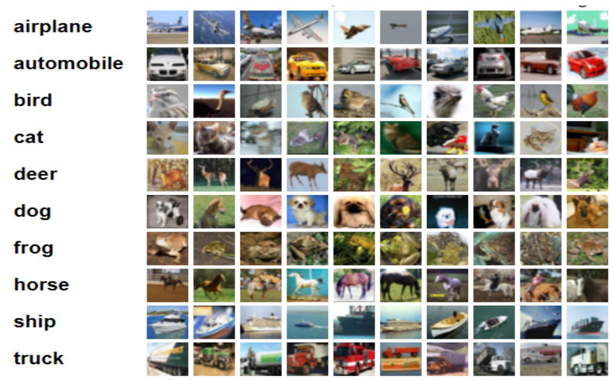


Fig. 1. CIFAR10 Dataset Sample Images

B. Proposed CNN Architecture

The proposed deep CNN architecture for this work is shown in the Fig. 2. It is a deep CNN with 6 convolutional layers, 3 maxpool layers, 1 flatten layer, and 1 output or fully connected layer. The 32x32 images from the CIFAR10 dataset are given to the first convolutional layer(Conv1). Each convolutional layer provided with 3 functions here, that are convolution, activation, and batch normalization. So that understand different characteristics of images. Here uses 3x3 sized filter, it creates many feature maps. Also set stride to one and padding to 'same' so that size of the matrix is not changed. Exponential linear unit(ELU) is the activation function used here to activate the neurons in the layer. Also batch normalization is used to stabilize the network and speed up the training process. This will normalize the activations from the previous activation layer. The third layer is the maxpool layer(Maxpool1), which uses 2x2 filter and reduces the input size to half by taking only the maximum values. A dropout of 0.2 is given after this maxpool layer. This is done to reduce the overfitting of the network.

Both Conv1, Conv2 and Maxpool1 layer has 32 nodes per layer. 3rd and 4th convolutional layer(Conv3 and Conv4) has the same functions as the previous convolution layer. After this there is a 6th layer which is maxpool layer(Maxpool2) and it reduces the input size to half just like the previous maxpool layer. A dropout of 0.3 is provided after this maxpool layer. These three layers has 64 nodes per layer. These layers followed by 5th and 6th convolutional layers(Conv5 and Conv6) and they also have the same function as the previous convolutional layer. 3rd maxpool layer(Maxpool3) after this, again reduces the dimensionality of input to half. These 3 layers have 128 nodes per layer. A droupout of 0.4 is provided after this maxpool layer. The output from this last maxpool layer given to the flatten layer. The main function of the flatten layer is to convert or flatten the output from the previous layer into a single-dimensional feature vector. And this feature vector is given to the fully connected output layer. The output layer consists of 10 nodes corresponding to 10 different classes

Authorized licensed use limited to: San Francisco State Univ. Downloaded on June 21,2021 at 18:20:17 UTC from IEEE Xplore. Restrictions apply.

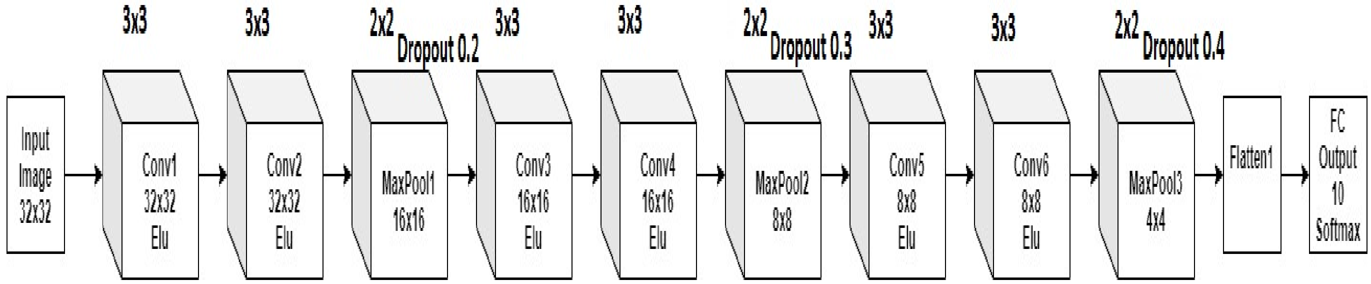


Fig. 2. Proposed CNN Architecture

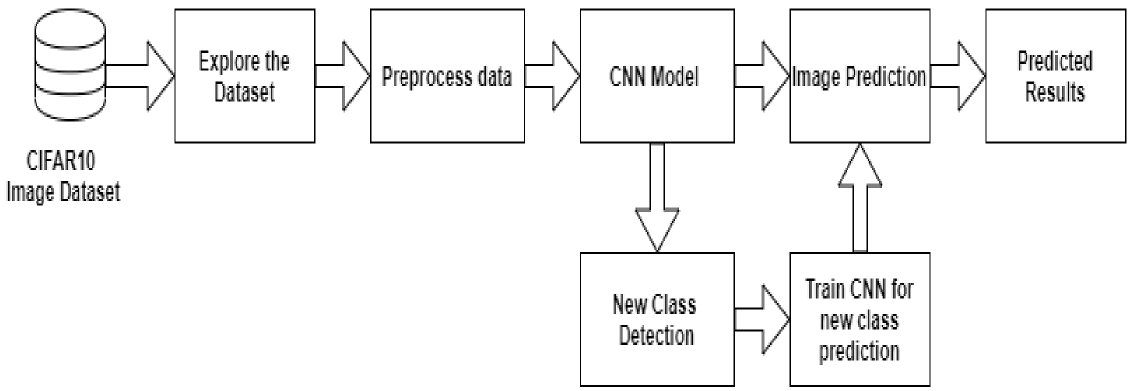


Fig. 3. Proposed Overall Architecture

of the CIFAR10 dataset. Softmax function in the output layer provides prediction scores or probabilities of the 10 different classes. The probabilities add up to one. The class label with the highest prediction score will be the predicted label for the input image by the network.

The build CNN is trained using CIFAR10 dataset. Here training dataset divided into 64 batches so that 781 steps in each epoch. Initially the learning rate set to 0.001, then for 75 above epoch it set to 0.0005 and 100 above epoch it set to 0.0003. The loss of the network is checked during the training. Because loss is the prediction error of the network, so needed to reduce the loss to get accurate prediction result. If the loss of the network is higher then during the next evaluation the weights updated to reduce the loss. The training and weight updation details are stored in model.h5 file. So during testing this model.h5 file loaded and make prediction results for the test data.

C. Implementation

Model implemented using python, tensorflow, and keras. Above discussed deep CNN architecture is created here. This deep structure of CNN helps to learn different features of images at different layers. Fig. 3. shows overall architecture of proposed work. Here need a basic understanding of the dataset like no of classes in the dataset, no of images for training and testing, etc these things are done at the explore dataset step. Better understanding of dataset makes better prediction results. Dataset converted to the required format for this work, which is done at prepocess step. More and more images are required for a deep CNN during training. For this data augmentation is used [8], this will increase the diversity of training data without collecting new images. This is done by performing different operations like rotate, crop, flop, shift, etc on training images. ImageDataGenerator is a class in keras, which is used to create data augmented batches of images in realtime. Training images are looped certain times to attain this. A different transformation like crop,shift, flip, zoom, etc are applied on the looped images, then finally get the output as the data augmented image data. In this way no of training images gets increased. Here using flow(x,y) method which can take numpy data and label arrays as the input. After that, they provide batches of augmented or normalized data as the output. The data augmentation done in this work are rotation range set to 15, width shift range set to 0.1, hight shift range set to 0.1, and horizontal flip set to True.

The next step is to train the created CNN with the CIFAR10 dataset. After training, during testing CNN makes prediction of labels for test images and shows the predicted results. So completion of these steps will create a better accurate classification model for CIFAR10 dataset. Then the next step is to detect the new class images. For this, provide a set of input images one after another which contains both existing class and new class images. This input continuously varying from existing class to new class image. When this input is given, then the output layer of CNN provides the softmax prediction score for different classes. Prediction score displays in the

Authorized licensed use limited to: San Francisco State Univ. Downloaded on June 21,2021 at 18:20:17 UTC from IEEE Xplore. Restrictions apply.

decreasing order. The class label with the highest prediction score will be the predicted class label for the input image. For existing class images the predicted label will be correct. But in the case of new class image it is not true. When look into the prediction score for different classes of a new class image, the highest prediction score for the first label is very very less when compared with existing class images highest prediction score. Even if the misclassified images of CIFAR10 dataset have a higher prediction score for the first class label. By looking into this prediction score trend understand that, for new class images the network is confused so that it cannot understand which class this input actually belongs to. So that the network classifies the new class image into any of the existing classes by considering the basic feature of images. Because of this the first class label itself has a very very less prediction score. In this work by looking into the first class label prediction score detect new class images.

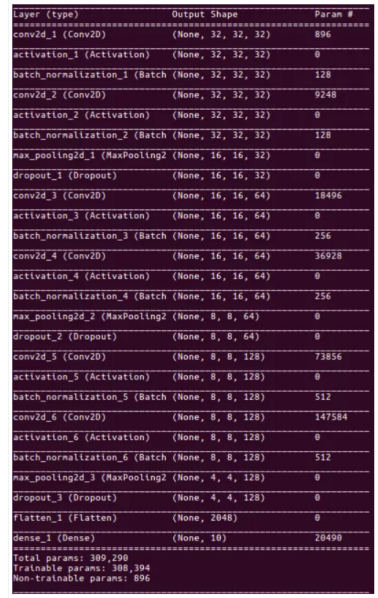
When a new class image is detected by this model, so in the future to classify these new images correctly, should add these new classes with the created classification model. For this add new class names to the class label list and add new nodes into the output layer of CNN according to new classes. Also 32x32 sized new class images are collected for training. Previous training and weight details are already with the model.h5 file. The new training and weight updation details are also added to this model.h5 file. So that during testing this file can be loaded and predict the new class correctly for new class images.

III. EXPERIMENTAL RESULTS

Fig. 4. shows the architecture model summary of CNN created for this work. The CNN trained for a different number of epochs. The classification accuracy is increased with increase in the epochs. Table 1. shows the classification accuracy obtained after training the CNN for different no of epochs. After CNN trained for 125 epochs the model reaches a classification accuracy of 89.87%. Some of the input test images prediction results given by this model shown in the Fig. 5.

TABLE I
CLASSIFICATION ACCURACY OF CNN

Epochs	Classification Accuracy
5	74.95%
20	83.92%
50	86.00%
125	89.87%



Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 32, 32, 32)	896
activation_1 (Activation)	(None, 32, 32, 32)	0
batch_normalization_1 (Batch	(None, 32, 32, 32)	128
conv2d_2 (Conv2D)	(None, 32, 32, 32)	9248
activation_2 (Activation)	(None, 32, 32, 32)	0
batch_normalization_2 (Batch	(None, 32, 32, 32)	128
max_pooling2d_1 (MaxPooling2	(None, 16, 16, 32)	0
dropout_1 (Dropout)	(None, 16, 16, 32)	0
conv2d_3 (Conv2D)	(None, 16, 16, 64)	18496
activation_3 (Activation)	(None, 16, 16, 64)	0
batch_normalization_3 (Batch	(None, 16, 16, 64)	256
conv2d_4 (Conv2D)	(None, 16, 16, 64)	36928
activation_4 (Activation)	(None, 16, 16, 64)	0
batch_normalization_4 (Batch	(None, 16, 16, 64)	256
max_pooling2d_2 (MaxPooling2	(None, 8, 8, 64)	0
dropout_2 (Dropout)	(None, 8, 8, 64)	0
conv2d_5 (Conv2D)	(None, 8, 8, 128)	73856
activation_5 (Activation)	(None, 8, 8, 128)	0
batch_normalization_5 (Batch	(None, 8, 8, 128)	512
conv2d_6 (Conv2D)	(None, 8, 8, 128)	147584
activation_6 (Activation)	(None, 8, 8, 128)	0
batch_normalization_6 (Batch	(None, 8, 8, 128)	512
max_pooling2d_3 (MaxPooling2	(None, 4, 4, 128)	0
dropout_3 (Dropout)	(None, 4, 4, 128)	0
flatten_1 (Flatten)	(None, 2048)	0
dense_1 (Dense)	(None, 10)	20490

Total params: 309,290
Trainable params: 308,394
Non-trainable params: 896

Fig. 4. CNN Model Summary

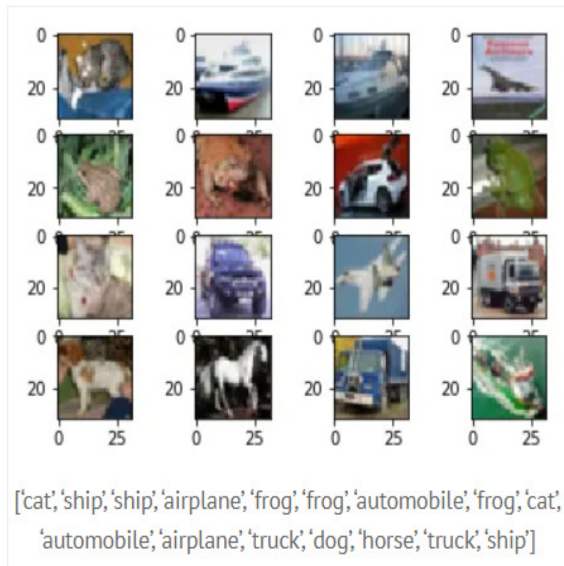


Fig. 5. Some of the Prediction Results

For new class detection two new classes of images are introduced here, that are flower and home class images. Inputting 5 images one after another in the order airplane(existing class), new class(flower), horse(existing class), new class(home), frog(existing class). The input images and its corresponding softmax predictions score for different class labels in the decreasing order shown in the Fig. 6. The first class label with the highest prediction score is the predicted class label for the image. For existing class images airplane, horse and frog the predicted class labels are correct. Flower and home images detected as new class here.

From these results observed that the prediction score for the first class label of a new class image is very very less. By testing different new class images one thing identified is that, most of the new class images have first class label prediction score is less than 0.3. Even if the existing class image is predicted as a wrong class label its first class label prediction score is higher than 0.3. Here detecting the new class images by looking into the first class label prediction score. If the first class label prediction is less than 0.3 then the input image is

Authorized licensed use limited to: San Francisco State Univ. Downloaded on June 21,2021 at 18:20:17 UTC from IEEE Xplore. Restrictions apply.

Fig. 6. Input Images and Corresponding Softmax Prediction Score

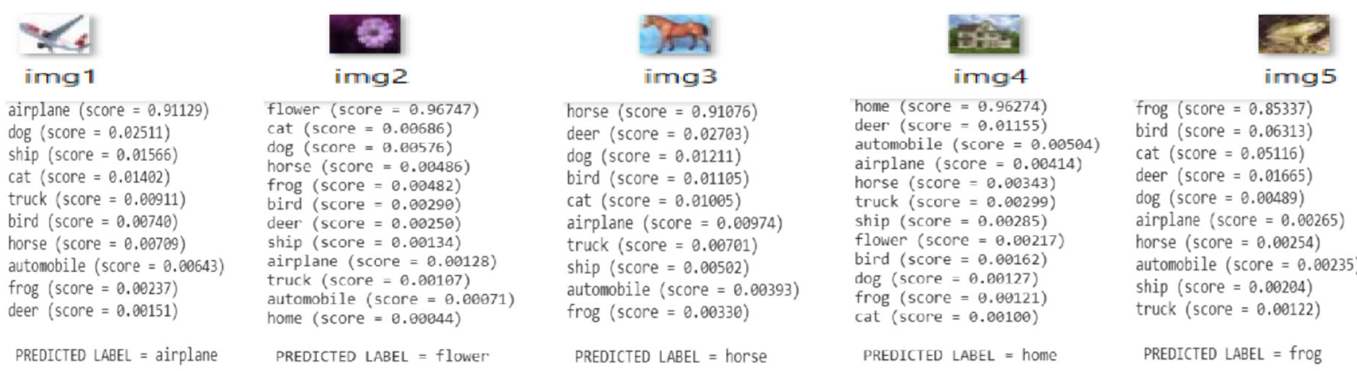


Fig. 7. Input Images and Corresponding Softmax Prediction Scores After Updation of New Class

considered as a new class image.

After the detection of two new classes(flower and home), collected images for these classes and trained the CNN with the newly collected images. These two class labels added with the class label list. Also extra two nodes added to the output layer of CNN to predict these new image classes. In this case not dividing training images to batches. In a single epoch all the collected images are trained. The training and weight updation details added to the previously stored model.h5 file. This updated model.h5 file is used for prediction during testing. The same set of 5 images inputted now. The images and its corresponding predictions score list are shown in the Fig. 7. New class images correctly predicted to flower and home respectively shown in this figure.

Table. 2. shows the classification accuracy of the proposed method compared with other CIFAR10 classification works discussed early. It shows that our model has a better classification accuracy than other discussed works.

TABLE II
COMPARING CLASSIFICATION ACCURACY BETWEEN OTHER METHODS AND PROPOSED METHOD

Methods	Classification Accuracy
FLSCNN	75.90%
PCANet	78.70%
DUFLCNN	82.00%
DCGANs	82.80%
PROPOSED METHOD	89.87%

IV. CONCLUSION

In this paper, proposed a CNN based image classification and new class detection model for CIFAR10 dataset. This model achieves approximately 90% classification accuracy. Image classification has higher importance in today's world. So there is always a need for achieving a good classification accuracy otherwise images get misclassified. Additionally here a new class detection method is proposed. Which is done by looking at the trends of softmax prediction score for the first class label of a new class image. Also this new class can be predicted correctly in the future by updating this model for new class prediction. Here created a better accurate classification and new class detection model for CIFAR10 dataset as an example. So in future this work can be considered as a base and extended for the classification and new class detection in other dataset and images in different applications. When comparing with other works that classify CIFAR10 dataset, this model has higher classification accuracy so this model gives better accurate prediction results.

Authorized licensed use limited to: San Francisco State Univ. Downloaded on June 21,2021 at 18:20:17 UTC from IEEE Xplore. Restrictions apply.

References

[1] Wang, Zhuoyi, Zelun Kong, Swarup Changra, Hemeng Tao, and Latifur Khan. ”Robust high dimensional stream classification with novel class detection.” In 2019 IEEE 35th International Conference on Data Engineering (ICDE), pp. 1418-1429. IEEE, 2019.

[2] Jmour, Nadia, Sehla Zayen, and Afef Abdelkrim. ”Convolutional neural networks for image classification.” In 2018 International Conference on Advanced Systems and Electric Technologies (ICASET), pp. 397-402. IEEE, 2018.

[3] Islam, Md Tohidul, BM Nafiz Karim Siddique, Sagidur Rahman, and Taskeed Jabid. ”Image recognition with deep learning.” In 2018 International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS), vol. 3, pp. 106-110. IEEE, 2018.

[4] McDonnell, Mark D., and Tony Vladusich. ”Enhanced image classification with a fast-learning shallow convolutional neural network.” In 2015 International Joint Conference on Neural Networks (IJCNN), pp. 1-7. IEEE, 2015.

[5] Chan, Tsung-Han, Kui Jia, Shenghua Gao, Jiwen Lu, Zinan Zeng, and Yi Ma. ”PCANet: A simple deep learning baseline for image classification?.” IEEE transactions on image processing 24, no. 12 (2015): 5017-5032.

[6] Dosovitskiy, Alexey, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. ”Discriminative unsupervised feature learning with convolutional neural networks.” In Advances in neural information processing systems, pp. 766-774. 2014.

[7] Radford, Alec, Luke Metz, and Soumith Chintala. ”Unsupervised representation learning with deep convolutional generative adversarial networks.” arXiv preprint arXiv:1511.06434 (2015).

[8] Shorten, Connor, and Taghi M. Khoshgoftaar. ”A survey on image data augmentation for deep learning.” Journal of Big Data 6, no. 1 (2019): 60.

Authorized licensed use limited to: San Francisco State Univ. Downloaded on June 21,2021 at 18:20:17 UTC from IEEE Xplore. Restrictions apply.