# Optimization of Convolutional Neural Network Parameters for Image Classification

Toshi Sinha, Brijesh Verma, and Ali Haidar
*Centre for Intelligent Systems*
*School of Engineering and Technology*
*Central Queensland University,* Australia
{t.sinha, b.verma,a.haidar}@cqu.edu.au

***Abstract*— Convolutional Neural Networks (CNNs) have been widely applied in image classification tasks. CNNs have a large number of parameters and they can produce different classification accuracy for same tasks based on diverse parameters including input window size, filter size, number of layers and number of neurons. The impact of these parameters on CNN accuracy in image classification tasks is investigated and analyzed in this study. A new methodology incorporating CNN for systematically conducting experiments to find the impact of diverse parameters is presented. Two datasets such as benchmark CIFAR-10 dataset and road-side vegetation dataset for real-world applications were selected to conduct this study. The experiments were conducted by varying different network parameters and recording the accuracy. Experimental analysis has shown that changing the number of layers and input window size has significant impact on classification accuracy of CIFAR-10, whereas for roadside vegetation dataset input window size and filter size have maximum impact on classification accuracy. The proposed optimization approach achieved higher accuracy (81%) than the accuracy obtained by Alexnet (77.75%) and PSO-CNN (80.15%) on CIFAR-10 dataset.**

***Keywords*— *Image Classification, Convolutional Neural Networks, Parameters Selection.***

## I. INTRODUCTION

Artificial neural networks have been investigated by computational intelligence researchers for many decades. Different neural network topologies have been proposed and applied in prediction and classification problems. Each network topology varies based on its architecture and links between its elements. Convolutional neural network is a deep learning neural network that models animal's visual cortex. CNN based models have been widely proposed to solve various computer vision problems including complex image segmentation and classification, object detection and optical character recognition [1].

Three types of layers are usually created to develop a CNN: convolutional layers, pooling layers and fully connected layers. Most of computation is done in the convolutional layer. The main power of a CNN lies in its deep architecture [1-3], which allows extracting a set of discriminating features at multiple levels of abstraction. Each of the convolution layer is composed of linear filtering (convolution), non-linearity and feature pooling stages [4]. Due to strong image recognition capabilities, in the past few years, CNNs are used across variety of applications including natural language processing, hyperspectral image processing and identifying different diseases in radiographic images [5, 6].

Krizhevsky et al. proposed that a large deep convolutional neural network is capable of achieving record breaking results on a highly challenging dataset using purely supervised learning [1]. The proposed architecture was named as AlexNet, it contained 8 learned layers among them 5 were convolutional and 3 were fully connected layers. To reduce overfitting, data augmentation and dropout were used [7]. Dropout was used in first 2 fully-connected layers. Two GPUs were used in parallel with a trick that GPUs communicate only in certain layers. 1.2 million high-resolution training images, 50,000 validation images and 150,000 test images were used in ImageNet LSVRC-2010. This architecture achieved top-1 and top-5 test set error rates of 37.5% and 17.0% respectively. Experiments prove that performance degrades by 2% if a single convolutional layer is removed. Here the impact of reducing one convolutional layer was tested but there were no experiments for increasing the number of layers, neurons or filter size.

Szegedy et al. proposed an inception architecture, which utilizes the computing resources inside the network more efficiently by increasing depth and width of the network [2]. Architectural decisions were based on Hebbian principle and intuition of multiscale processing. GoogLeNet is incarnation of such architecture. The network has 22 layers with parameters and 5 layers of pooling. All the convolutions including the inception modules used ReLU activation. In this architecture, auxiliary classifiers were connected to intermediate layers. Around 1.2 million images for training, 50,000 for validation and 100,000 images for testing were used. The finalist in ILSVRC 2014 challenge obtained a top-5 error of 6.67% on both validation and test data, ranking GoogLeNet as first among other participants. It was strongly suggested that moving to sparser architecture is feasible and useful in improving neural network for computer vision. Even though this architecture used fewer number of parameters but the inherent relationships between network configurable parameters, their effect on overall accuracy was not well understood and analysed.

Srivastava et al. presented a new regularization technique named as "Dropout", to reduce the overfitting problem in neural networks [7]. This is achieved by randomly dropping

978-1-5386-2726-6/17/$31.00 ©2017 IEEE

half of the feature detectors on every training case. The feature detectors which are dropped out in this way do not participate in forward pass and do not contribute to back-propagation. Due to this, complex co-adaptation of neurons are reduced and each hidden units are encouraged to learn a robust feature as it cannot rely on other hidden unit to correct its error. They performed experiments on CNN like ImageNet [1] using MNIST dataset [4], CIFAR-10 dataset [8] and demonstrated that "Dropout" forces models to acquire robust features which are less co-adapted and hence better results are achieved in image and speech recognition.

Tajbakhsh et al. proposed that pre-trained deep CNN with sufficient fine tuning eliminates the need for training a deep CNN from scratch [5]. Acquiring large amount of labeled data in the medical field can be challenging. AlexNet model was used because this architecture was deep enough to investigate the impact of the depth of fine-tuning on the performance of pre-trained CNNs. The experiments were carried on following 4 category images: P olyp Detection, Pulmonary Embolism Detection, Colonoscopy Frame Classification and Intima-Media Boundary Segmentation. It was found that, fine tuning a pre-trained CNN layer-wise manner leads to incremental performance improvement. Also, the performance gap between deeply fine-tuned CNNs and those trained from scratch widened, when the size of training sets was reduced. The paper concludes that fine-tuned CNNs should always be the preferred option regardless of the size of training sets available. Still there was no evidence of changing input image size or filter size can affect the accuracy, all emphasis was given on training from scratch or fine tuning.

Sun et al. proposed an excellent idea of semi-supervised deep convolutional neural network for breast cancer diagnosis, which can use large amount of unlabelled data along with a small amount of labelled data [6]. Due to the confidentiality agreements, acquiring labelled data in medical field is difficult task. Neither the doctors nor the patients are willing to share the information, whereas there is abundance of unlabelled data. Also, it needs several radiologists to evaluate each data individually and then compare the results with peer, case by case to reach to a c onclusion, called as label. In such social circumstances, this approach is remarkable, as it uses a very small number of labelled images and rest of the images (data) are unlabelled to train the CNN. Four modules were used in this diagnostic system: data weighing, feature selection, dividing co-training data labelling, and CNN. 3158 Region of Interests (ROIs) with each containing a mass extracted from 1874 pairs of mammogram images were used for this study. Among them 100 ROIs were treated as labelled data while the rest were treated as unlabelled. The highest accuracy of CNN was 0.8243 using the mixed labelled (100) and unlabelled data (1774).

LeCun et al. proposed the end to end training of deep neural network using gradient based optimization [4]. They were among the very first and few researchers, who strongly recommended that better pattern recognition systems can be built by relying more on automatic learning than hand designed heuristics. In this paper, various methods were tested for hand written character recognition and compared with standard hand written digit recognition tasks. It was proved that, in particular CNN can be combined with search mechanism or inference mechanism to model intricate outputs that are interdependent. A good example could be document recognition and hand written character recognition. An example was demonstrated and named as Graph Transformer Network (GTN), it is multi-layer and multi module network where a graph is used to represent the state information, instead of fixed size vectors. GTN is used for reading bank check, it uses CNN character recognizer combined with global training techniques to provide accuracy on checks. Currently, it is deployed commercially and it reads millions of checks per day. Even though the paper talks about optimization but there was no study done on the network parameters and their optimization.

Talathi proposed a Sequential Model Based Optimization (SMBO) strategy to optimize the hyper-parameter of deep convolutional network for object identification [9]. CIFAR-10 benchmark dataset was used for testing purposes. It was proved that SMBO can be used to generate many good/superior deep convolutional network architectures. But there was no evidence to suggest the tuning of hyper-parameters or the impact of each parameter.

The overall aim of this research is to examine the effect of deep learning parameters on accuracy and optimise them. The impact of convolutional neural network parameters on accuracy is investigated in this paper. The remainder of this paper consists of four sections. Section II describes the proposed approach. Section III presents datasets used for experiments. Section IV presents experiments and results. Section V concludes the paper.

## II. Proposed Approach

The proposed approach for analysis and optimization of network parameters in deep learning based convolutional neural networks is shown in Fig. 1 and described by Equation 2. In this research, CNN parameters as described below are systematically varied and the classification accuracies are recorded.

- Input image size ($W_x \times H_x \times D_p$)
- Filter size $(F_y)$, the receptive field size of first convolutional layer
- Number of neurons $(K_z)$, in the first convolutional layer
- Number of layers, M and N

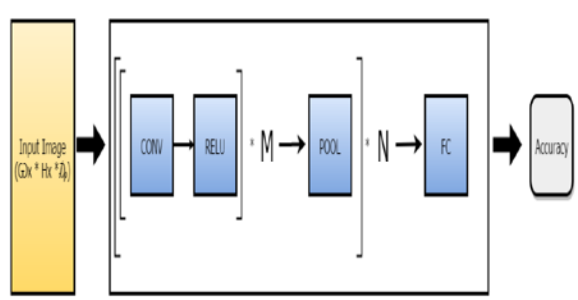


Fig. 1. Proposed Approach.

In the proposed design, the CNN architecture stacks a few CONV - RELU layers (e.g. M number of layers), followed by pooling layers (optional). This pattern is repeated N times until the image has been merged spatially to a small size. At the end, there is a Fully-Connected (FC) layer to hold the output called as class score. Each neuron, of each layer computes following activation function.

$$f\ (x) = \varphi(w^T\ x\ +\ b) \qquad (1)$$

where $x$ is the input to the neuron, $w$ is a weight vector, $b$ is a bias term and φ is a nonlinear function. Each neuron multiple inputs and produces a single output.

The convolutional layer is the main building block of a CNN; it consists of a set of learnable filters. Every filter is spatially small (spans along width and height), but extends through full depth of input volume. In forward pass, each filter slides across width and height of input volume and generated dot product between entries of the filter and input at the position, this generates a 2-D activation map which gives the responses of that filter at every spatial position. These 2-D activation maps are stacked along depth and output volume is produced. So, the output volume can be stated as a function of input volume as follows.

An image having width $W$, height $H$ and depth $D_p$ color channels (i.e. $W \times H \times D_p$), the learnable filter divides the image width as $W_1 = (W - F + 2P)/(S + 1)$ where $F$ refers to the spatially extent neuron size, $P$ is the amount of zero padding, and $S$ is the size of stride. Similarly, the height is divided by $H_1 = (H - F + 2P)/(S + 1)$, depth $D_p$ is the size of number of filters $K$.

In proposed approach, the input image varies by the $x$ factor. This means the width and height vary but the depth is constant $(W_x \times H_x \times D_p)$. Also, the filter size, known as spatially extent neuron size $F_y$ changes. Hence the first filter of first CONV layer divides the image width as $W_1 = \left(W_x - F_y + 2P\right)/(S + 1)$ and $H_1 = \left(H_x - F_y + 2P\right)/(S + 1)$ and $D_{p1} = K_z$.

The behaviour of CNN is very complex and the performance is heavily dependent on many parameters. In this research, the main aim is to maximize the image classification accuracy (A), which can be represented as a f unction that varies due to variation in the stated internal CNN parameters:

$$A = f\left(I_x, F_y, K_z, L\right) \qquad (2)$$

Where $I_x$ represents the Input image size $(W_x \times H_x \times D_p)$, $F_y$ filter size, $K_z$ number of filters, L number of layers.

The very first step is to determine the candidate value for input image size, filter size, number of filters and number of layers.

For input image size the candidate values are {16x16, 28x28, 32x32, 64x64 and 128x128} the depth is 3 for all input images as it denotes RGB. By keeping rest of the parameters at a fix value the CNN model is trained and tested and the classification accuracy is recorded. Further if there is a sharp decrease in classification accuracy due to a particular input image size, that candidate is dropped out from the experiments.

Secondly, for filter size the candidate values are {1x1, 3x3, 5x5, and 7x7}. Now in this step all the training and testing is carried out with the variation in both input size and filter size, and classification accuracy is recorded. Note that the candidates which have performed badly or dropped out in first step do not participate in this step.

Thirdly, for the number of filters the candidate values are {32, 28, and 20}. Again all the training and testing is carried out and classification accuracy is calculated with all combination of input size, filter size and number of filters. Also, the candidates that were dropped out in first and second step do not participate in this.

Lastly, for number of layers only 2 models were taken into consideration {11 and 13 layered}, which are shown in Table II and Table III. These 2 models are extensively trained and tested with different combination of internal parameters and the classification accuracy is recorded.

## III. Data

Two datasets were used in this study: CIFAR-10 and road-side vegetation datasets. CIFAR-10 is a benchmark dataset that is widely applied in computer vision applications. CIFAR-10 was collected from [8]. The dataset consists of 60,000 images in 10 object categories called as classes. The main reason behind taking this dataset is that it has real world examples with a wide variety and range for each class. Fig. 2 shows sample image classes of CIFAR-10 data.

Road-side vegetation dataset is a real-world image data that contains seven different classes (brown grass, green grass, road, sky, soil, tree-leaf and tree-stem). This dataset composed of 653 images, can be considered a small dataset. Road-Side Vegetation (RSV) dataset was taken from [10]. Figure 3 shows classes of road-side vegetation dataset.

## IV. Experiments and Results

The proposed approach was implemented and experiments were conducted. Two main phases are followed in this study: training and testing. The training phase uses input images with labelled output to let the network learn. In testing phase, the trained model is evaluated. Ten-fold cross validation was used.

The parameters that were investigated in this study were: input image size, filter size, number of neurons and number of layers. A number of experiments were conducted by varying the network parameters. The network was fine-tuned with each of these elements and accuracy was recorded. The percentage accuracy was calculated using the following equation.

$$Percentage\ Accuracy = (1 - top1err) \times 100 \qquad (3)$$

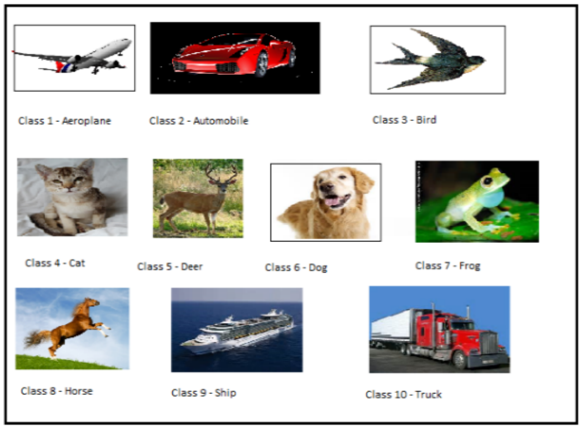


Fig. 2. Sample image classes of CIFAR-10 dataset.


Fig. 3. Sample image classes of RSV dataset.

where $top1err$ is the deviation of the top class from the target label. The accuracy was calculated over both training and validation datasets.

Input image sizes 16×16×3, 28×28×3, 32×32×3 and 64×64×3 were used. Then, filter size was modified and applied with different input image size. Furthermore, number of filters were altered. Finally, numbers of layers were changed with all parameters being changed. The values used in this study with each parameter are shown in Table I. Due to change in number of layers, following two CNN architectures as shown in Tables II and III are being created and tested.

The recorded accuracies for each modification are shown in Table IV. The accuracies on training dataset varied between 35% and 84.5% for roadside vegetation dataset and between 59.9% and 95.8% for CIFAR-10 dataset. Accuracies on testing dataset ranged between 38% and 87% for roadside vegetation dataset and between 57.5% and 81% for CIFAR-10 dataset.

Fig. 4 shows the accuracy graph for CIFAR-10 dataset with different input image sizes. Fig. 5 shows system accuracy graph for RSV dataset. The highest accuracy for RSV was achieved with 64x64x3 input window size. The highest accuracy obtained for CIFAR-10 was with 32x32x3 input window size. It is clearly shown that increasing input image size increased overall performance with RSV dataset. It can be stated that for both the datasets accuracy increases by increasing the input image size till a saturation point (Imax) is reached, once this point is reached, accuracy starts declining by increasing the input image size. In other words accuracy follows a positive gradient with increasing input image size and attends to maximum before turning to negative gradient. For CIFAR-10 the highest accuracy was achieved at 32x32x3, whereas for road-side vegetation dataset highest accuracy was obtained with 64x64x3 input image size.

TABLE I. PARAMETERS.

| Parameter | Values |
|---|---|
| Input Image size | 16×16×3<br>28×28×3<br>32×32×3<br>64×64×3 |
| Filter size | 3×3<br>5×5<br>7×7 |
| Number of Filters | 28<br>32 |
| Layers | 11<br>13 |

TABLE II. PROPOSED ARCHITECTURE (11 LAYERS).

| Block | Number | Layer type |
|---|---|---|
| Block1 | L1 | Conv |
| | L2 | Pool |
| | L3 | Relu |
| Block2 | L4 | Conv |
| | L5 | Relu |
| | L6 | Pool |
| Block3 | L7 | Conv |
| | L8 | Relu |
| | L9 | Pool |
| Block 4 | L10 | Conv |
| Loss Layer | L11 | SoftMax |

TABLE III. PROPOSED ARCHITECTURE (13 LAYERS).

| Block | Number | Layer type |
|---|---|---|
| Block1 | L1 | Conv |
| | L2 | Pool |
| | L3 | Relu |
| Block2 | L4 | Conv |
| | L5 | Relu |
| | L6 | Pool |
| Block3 | L7 | Conv |
| | L8 | Relu |
| | L9 | Pool |
| Block4 | L10 | Conv |
| | L11 | Relu |
| Block5 | L12 | Conv |
| Loss Layer | L13 | SoftMax |

TABLE IV. ACCURACY OBTAINED ON ROADSIDE-VEGETATION DATA AND CIFAR -10 DATA WITH DIFFERENT COMBINATIONS OF ALL FOUR PARAMETERS.

| **CNN Parameters** | | | | **Roadside Vegetation Dataset** | | **CIFAR - 10 Dataset** | |
|---|---|---|---|---|---|---|---|
| Input image size | Filter size | Number of filters | Number of layers | Accuracy (Training) | Accuracy (Testing) | Accuracy (Training) | Accuracy (Testing) |
| 16 x 16 x 3 | 3 x 3 | 28 | 11 | 73.1% | 74.7% | 75.7% | 69.9% |
| | | 28 | 13 | 76.0% | 77.2% | 81.3% | 73.4% |
| | | 32 | 11 | 74.7% | 76.5% | 76.6% | 70.0% |
| | | 32 | 13 | 77.2% | 77.2% | 81.2% | 73.5% |
| | 5 x 5 | 28 | 11 | 74.7% | 73.5% | 77.9% | 71.3% |
| | | 28 | 13 | 73.9% | 72.2% | 81.9% | 73.3% |
| | | 32 | 11 | 74.7% | 74.1% | 78.1% | 70.9% |
| | | 32 | 13 | 73.5% | 72.8% | 82.6% | 73.8% |
| | 7 x 7 | 28 | 11 | 68.6% | 67.3% | 72.5% | 65.9% |
| | | 28 | 13 | 77.2% | 77.2% | 81.4% | 72.6% |
| | | 32 | 11 | 68.8% | 67.9% | 73.7% | 65.9% |
| | | 32 | 13 | 77.4% | 75.9% | 81.9% | 72.5% |
| 28 x 28 x 3 | 3 x 3 | 28 | 11 | 77.4% | 80.2% | 85.5% | 76.1% |
| | | 28 | 13 | 80.0% | 81.5% | 89.3% | 78.5% |
| | | 32 | 11 | 77.8% | 81.5% | 85.5% | 76.9% |
| | | 32 | 13 | 77.2% | 77.8% | 91.4% | 78.6% |
| | 5 x 5 | 28 | 11 | 78.8% | 80.9% | 84.0% | 76.1% |
| | | 28 | 13 | 72.7% | 74.1% | 91.2% | 78.2% |
| | | 32 | 11 | 77.4% | 80.2% | 87.0% | 77.6% |
| | | 32 | 13 | 64.2% | 71.0% | 88.4% | 77.7% |
| | 7 x 7 | 28 | 11 | 74.9% | 75.3% | 86.6% | 76.7% |
| | | 28 | 13 | 71.7% | 69.8% | 89.7% | 76.9% |
| | | 32 | 11 | 78.2% | 77.8% | 84.6% | 75.6% |
| | | 32 | 13 | 69.0% | 73.5% | 84.5% | 76.1% |
| 32 x 32 x 3 | 3 x 3 | 28 | 11 | 78.6% | 83.3% | 78.0% | 70.8% |
| | | 28 | 13 | 77.0% | 75.9% | 94.6% | 80.7% |
| | | 32 | 11 | 81.9% | 85.2% | 78.0% | 70.6% |
| | | 32 | 13 | 70.0% | 72.0% | 93.7% | 80.5% |
| | 5 x 5 | 28 | 11 | 79.4% | 80.2% | 81.2% | 72.9% |
| | | 28 | 13 | 44.4% | 44.4% | 95.8% | **81.0%** |
| | | 32 | 11 | 81.5% | 83.3% | 81.4% | 72.8% |
| | | 32 | 13 | 66.0% | 68.0% | 94.7% | 80.6% |
| | 7 x 7 | 28 | 11 | 76.6% | 77.2% | 82.1% | 73.5% |
| | | 28 | 13 | 67.0% | 65.4% | 94.6% | 80.6% |
| | | 32 | 11 | 78.4% | 77.2% | 82.2% | 73.8% |
| | | 32 | 13 | 66.0% | 65.0% | 92.8% | 79.9% |
| 64 x 64 x 3 | 3 x 3 | 28 | 11 | 78.0% | 85.0% | 85.1% | 75.6% |
| | | 28 | 13 | 83.9% | 85.8% | 86.4% | 75.7% |
| | | 32 | 11 | 84.5% | **87.0%** | 84.9% | 75.3% |
| | | 32 | 13 | 80.0% | 84.6% | 85.6% | 75.6% |
| | 5 x 5 | 28 | 11 | 78.2% | 84.6% | 85.7% | 76.4% |
| | | 28 | 13 | 45.6% | 44.4% | 59.9% | 57.5% |
| | | 32 | 11 | 70.5% | 71.6% | 85.4% | 77.3% |
| | | 32 | 13 | 63.0% | 64.0% | 90.5% | 78.1% |
| | 7 x 7 | 28 | 11 | 72.9% | 71.6% | 81.8% | 75.5% |
| | | 28 | 13 | 35.0% | 38.0% | 89.3% | 77.9% |
| | | 32 | 11 | 71.3% | 68.5% | 84.7% | 77.2% |
| | | 32 | 13 | 40.0% | 40.0% | 80.2% | 74.3% |

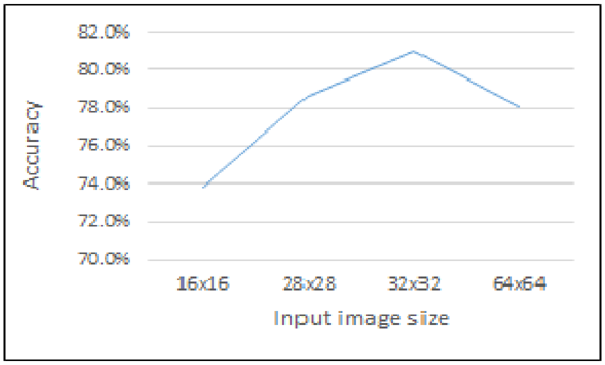


Fig.4 Accuracy graph for CIFAR-10 dataset for different input size

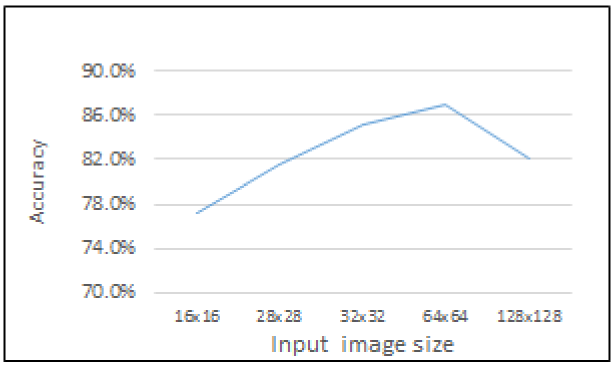


Fig.5 Accuracy graph for road-side vegetation dataset for different input size.

For roadside vegetation dataset, the highest test accuracy was obtained with input image size 64×64×3 and filter size 3×3. The number of filters and layers had least impact on system accuracy. Increasing image input size increased the accuracy of

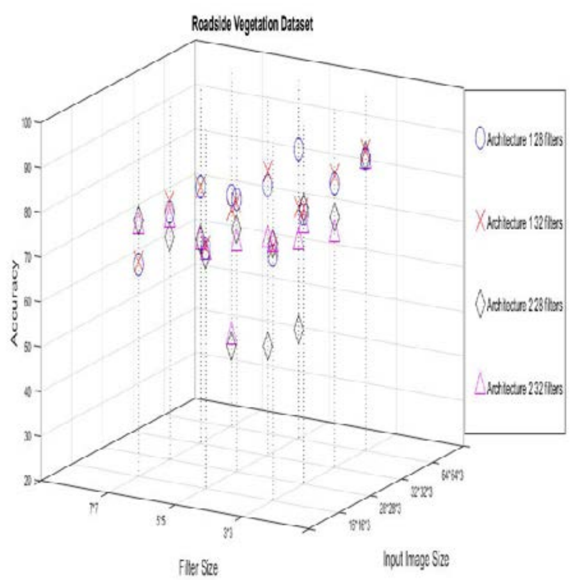


Fig. 6 Accuracy graph for road-side vegetation dataset with all four parameters.

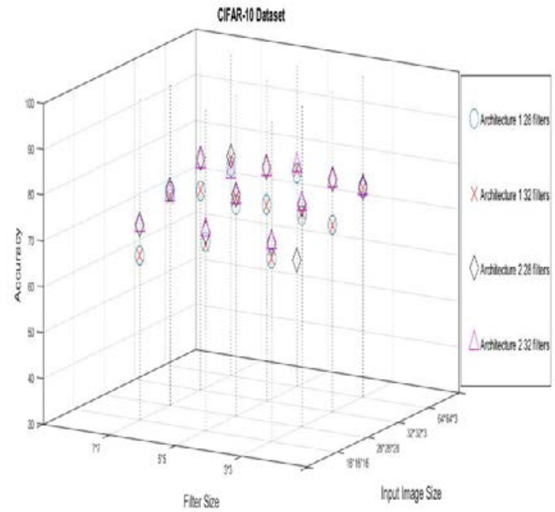


Fig. 7 Accuracy graph for CIFAR-10 dataset with all four parameters.

convolutional neural network performance over road-side vegetation dataset. Whereas for CIFAR-10 dataset, number of layers had immense impact on test accuracy. The highest test accuracy was obtained with input image size 32×32×3 and architecture 2 (13 layers).

The experimental results have been analysed and shown in Fig. 6 and Fig. 7. In the case of the CIFAR-10 dataset, it is very clear that increasing the number of layers increases the accuracy. The best accuracy was obtained using Architecture 2, having number of layers 13, optimal input window size 32×32×3, filter size 5x5 and the smaller number of filters e.g. 28. In the case of vegetation data set, the best system accuracy was achieved with Architecture 1, having number of layers 11, optimal input window size 64×64×3 and higher number of filters e.g. 32. The smaller filter size 3×3 produced a higher accuracy.

The results obtained using the proposed approach have been compared with recently published results [11] on CIFAR-10 dataset. The proposed approach produced significantly higher accuracy than the Alexnet [11] and slightly higher accuracy than the PSO-CNN [11].

## V. Conclusion

The impact of changing convolutional neural network parameters on classification accuracy was investigated in this study. Input image size, filter size, number of neurons and number of layers were systematically varied in an attempt to understand their impact on classification accuracy. Two datasets were used to perform the study. The numbers of layers and input window size have significant impact on accuracy for CIFAR-10 dataset. The highest classification accuracy of 81% was obtained with input image size 32×32×3, number of filters 28, filter size 5x5 and 13 layers, which is better than the best accuracy obtained by standard Alexnet classification accuracy of 77.75% on CIFAR-10 dataset. Whereas for roadside vegetation dataset input window size and filter size has maximum impact on classification accuracy. Highest system accuracy 87% was obtained with input image size 64×64×3, filter size 3×3 and 11 layers. The accuracy trend in both the graphs shown in Fig. 6 and Fig. 7 is not uniform, which means that varying network parameters have different impact on large and small datasets. Whereas by looking at the similar accuracy trend of graphs in Fig. 4 and Fig. 5, it is clear that by increasing input image size, the classification accuracy increases to reach a maximum accuracy point and then it declines on further increase of input image size, so it can be considered as optimal input image size. For every dataset this optimal input image size can be different. In the future research, the experiments on more datasets will be conducted and analysed. The evolutionary computation based optimisation algorithms will also be investigated and compared.

## References


[1] 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.

[2] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, *et al.*, "Going deeper with convolutions," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1-9.

[3] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, 2009, pp. 248-255.

[4] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proceedings of the IEEE, vol. 86, 1998, pp. 2278-2324.

[5] N. Tajbakhsh, J. Y. Shin, S. R. Gurudu, R. T. Hurst, C. B. Kendall, M. B. Gotway, *et al.*, "Convolutional neural networks for medical image analysis: full training or fine tuning?," IEEE transactions on medical imaging, vol. 35, 2016, pp. 1299-1312.

[6] W. Sun, T.-L. B. Tseng, J. Zhang, and W. Qian, "Enhancing deep convolutional neural network scheme for breast cancer diagnosis with unlabeled data," Computerized Medical Imaging and Graphics, vol. 57, 2017, pp. 4-9.

[7] 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, 2014, pp. 1929-1958.

[8] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," Computer Science Department, University of Toronto, Technical Report, 2009.

[9] S. S. Talathi, "Hyper-parameter optimization of deep convolutional networks for object recognition," in 2015 IEEE International Conference on Image Processing (ICIP), 2015, pp. 3982-3986.

[10] L. Zhang, B. Verma, and D. Stockwell, "Class-semantic color-texture textons for vegetation classification," in International Conference on Neural Information Processing, 2015, pp. 354-362.

[11] T. Yamasaki, T. Honma, and K. Aizawa, "Efficient optimization of convolutional neural networks using particle swarm optimization", IEEE Third International Conference on Multimedia Big Data, 2017, pp. 70-73.