

Deep Learning: Parameter Optimization Using Proposed Novel Hybrid Bees Bayesian Convolutional Neural Network

Nawaf Mohammad H Alamri , Michael Packianather , and Samuel Bigot 

School of Engineering, Cardiff University, Cardiff, UK

ABSTRACT

Deep Learning (DL) is a type of machine learning used to model big data to extract complex relationship as it has the advantage of automatic feature extraction. This paper presents a review on DL showing all its network topologies along with their advantages, limitations, and applications. The most popular Deep Neural Network (DNN) is called a Convolutional Neural Network (CNN), the review found that the most important issue is designing better CNN topology, which needs to be addressed to improve CNN performance further. This paper addresses this problem by proposing a novel nature inspired hybrid algorithm that combines the Bees Algorithm (BA), which is known to mimic the behavior of honey bees, with Bayesian Optimization (BO) in order to increase the overall performance of CNN, which is referred to as BA-BO-CNN. Applying the hybrid algorithm on Cifar10DataDir benchmark image data yielded an increase in the validation accuracy from 80.72% to 82.22%, while applying it on digits datasets showed the same accuracy as the existing original CNN and BO-CNN, but with an improvement in the computational time by 3 min and 12 s reduction, and finally applying it on concrete cracks images produced almost similar results to existing algorithms.

ARTICLE HISTORY

Received 15 October 2020
Revised 15 December 2021
Accepted 18 January 2022

Introduction

Artificial Intelligence (AI) builds on the idea of making machines behave like humans, facilitating the development of intelligent systems (Li et al. 2017) in order to increase the productivity and maximize the efficiency of the processes such as manufacturing machines. The most popular AI techniques are based on Artificial Neural Network (ANN), a type of Machine Learning inspired by biological human brain. These computing systems can be used to model big data and find complex relationships (De Filippis et al. 2017). ANNs have the ability to handle high-dimensional real-time data and extract implicit meaningful patterns that can be used to predict the future state of complex systems (Wuest et al. 2016). In addition, ANNs are capable of handling complex dynamic problems due to its ability to deal with

CONTACT Nawaf Mohammad H Alamri  alamrinm@cardiff.ac.uk  School of Engineering, Cardiff University, Cardiff CF24 3AA, UK

© 2022 The Author(s). Published with license by Taylor & Francis Group, LLC.
This is an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

nonlinearity. It is worth noting that ANNs are trained on historic data with relative ease by adjusting its control parameters such as learning rate and momentum.

Big Data analytics (Geissbauer, Vedso, and Schrauf 2016) is an integral part of the industry 4.0 paradigm, known as the fourth industrial revolution, which aims at creating smart systems where technologies are transformed by Internet of Things (IoTs), Cyber-Physical Systems (CPSs) and Cloud Computing (CC). Modeling an IoT system is based on modeling stochastic system addressing the relationship between process and system performance and providing a quantitative analysis of system performance (Ciortea 2018, August). On the other hand, there are many challenges remaining when applying ANN (Wuest et al. 2016), the biggest being data acquisition as the availability of relevant data are not guaranteed. In addition, after collecting a dataset, applying appropriate data mining can also be challenging, particularly for cases where a high amount of irrelevant data may have been collected, thus affecting the performance of the produced models.

As an extension to ANN capabilities, Deep Learning (DL) techniques are now well established, producing better learning capability as stated by Singh et al. (2018), they have the advantage of using automatic feature extraction by learning large numbers of nonlinear filters before the decision-making stage. One of the most popular DL networks is Convolutional Neural Network (CNN) (Singh et al. 2018).

This paper addresses designing better topology issue by proposing novel nature inspired hybrid algorithm that combines the Bees Algorithm (BA), which is known to mimic the behavior of honey bees (Ebubekir 2010), with Bayesian Optimization (BO) in order to increase the overall performance of CNN which is referred to as BA-BO-CNN, so the contribution of this study is improving the performance of CNN using novel hybrid BA-BO-CNN algorithm.

The paper is structured as follows: [section II](#) shows a general review of DL followed by that of CNN in particular is presented in [section III](#) together with the gaps and open issues that need to be addressed in order to improve the performance of CNN models. [Section IV](#) shows the impact of integration nature inspired algorithms with DL networks, in addition it proposes a novel hybrid BA-BO-CNN algorithm, [section V](#) presents the results and discussion. Finally, [section VI](#) concludes the study and suggests future direction.

Deep Learning

This section introduces the main definition, advantages, limitations, and applications of DL.

Definition

DL is a class of machine-learning techniques (Singh et al. 2018) that utilizes multiple processing layers where the output from a previous layer is used as an input for the following layer, to learn representations of data with multiple levels of abstraction. The difference between traditional learning and DL can be seen in terms of the feature extraction process, which depends on the data and the model types. Using the traditional learning leads to significant time being consumed while applying a trial-and-error approach to the feature extraction process and the success will depend on the user experience. On the other hand, the DL approach will benefit from an automatic feature extraction process through the learning of a large number of nonlinear filters before making decisions. Thus, DL combines feature extraction and decision-making within one model and avoids the often-suboptimal manual handcrafting.

Advantages, Limitations, and Applications of Deep-Learning Networks

DL can have many network structures that can then be used in different real-life applications. Each network will have its own advantages, limitations depending on the applications as shown in Table 1. These applications are coming from many fields, such as manufacturing. Wang et al. (2018) mentioned that DL enables advanced analytics for smart manufacturing systems in terms of object, equipment, process, people, and environment using aggregated big data. DL can help describe what happened in a process by capturing products' conditions and then by supporting the diagnoses of why particular issues occurred, examining the causes and detecting specific failures. After that DL can be used to help predict what will happen, for example predicting products' quality deviations, and finally to assist in the prescription of corrective actions by identifying measures to be taken in order to improve the quality of a product. The deep insights brought by DL can support a company's decision-making process throughout a product lifecycle by improving analyses of data emerging from design, manufacturing, and the supply chain, enhancing the process control and shorten downtime. DL has been applied in a wide range of manufacturing systems specially in the area of fault diagnosis and product quality inspection.

In addition to the manufacturing context, there are other fields in which DL has great applications. DL has been applied in the pharmaceutical context (Ekins 2016), such as to predict aqueous solubility, the epoxidation site in molecules, liver injuries induced from drugs as well as to diagnose cancer, to extract pattern in gene expression, to predict protein disorder, to analyze the content of breast cancer, to repurpose drugs and for microscope images classification. Thus, while DL appears to be a promising tool in biological context it is generally expected to have even greater applications in the future.



Table 1. Deep learning networks.

Deep Learning Network	Advantages	Limitations	Applications	References
Convolutional Neural Network (CNN)	<p>Eliminating the needs for extracting the features manually, it is achieved by alternating and stacking pooling and convolutional layers that convolve with the input data to extract the most significant features</p> <p>Can be used with image data, text data, time series data and sequence input data networks</p> <p>Has many pretrained architectures like AlexNet, GoogLeNet, ZFNet, RESNet and VGGNet</p> <p>Reducing the number of parameters</p> <p>Making directed cycle so that the outputs do not only depend on the immediate previous input</p> <p>Can be used with text data, speech data, time Series data, tabular data and generative models</p> <p>Allowing the information to persist in hidden layer and capturing the previous states and temporal correlation</p>	<p>High computational complexity in model training</p>	<p>Image, pattern, and speech recognition</p> <p>Image and sentence classification</p> <p>Prediction problems</p> <p>Regression problems</p>	<p>(Singh et al. 2018)</p> <p>(Wang et al. 2018)</p> <p>(MathWorks-1)</p> <p>(Tch 2017)</p>
Deep Recurrent Network (DRN)	<p>Reducing the number of parameters</p> <p>Making directed cycle so that the outputs do not only depend on the immediate previous input</p> <p>Can be used with text data, speech data, time Series data, tabular data and generative models</p> <p>Allowing the information to persist in hidden layer and capturing the previous states and temporal correlation</p>	<p>Cannot be used with image data</p> <p>Difficult to memorize the long-term interaction in training</p>	<p>Speech and connected handwriting recognition</p> <p>Classification problems</p> <p>Regression prediction problems</p>	<p>(Wang et al. 2018)</p> <p>(Tch 2017)</p> <p>(Davydova 2017)</p>
Sequence to Sequence Model (S2SM)	<p>It is a model with two recurrent networks, including encoder to process the input and decoder to produce the output</p>	<p>It is limited to sequence problems</p>	<p>System for answering question and machine translation</p>	<p>(Tch 2017)</p> <p>(Davydova 2017)</p>
Long Short-Term Memory (LSTM)	<p>Modeling temporal sequences along with their long-term dependencies more accurate than traditional recursive neural network</p>	<p>There is no activation function and no modification for the stored values</p>	<p>Speech recognition</p>	<p>(Tch 2017)</p> <p>(Davydova 2017)</p>
Gated Recurrent Network (GRN)	<p>Modeling temporal sequences along with their long-term dependencies more accurate than traditional recursive neural network</p> <p>It is similar to long short-term memory but without output gate</p>	<p>There is no activation function and no modification for the stored values</p>	<p>Speech recognition</p>	<p>(Tch 2017)</p>
Deep Belief Network (DBN)	<p>Inferring the states of the unobserved variables</p> <p>Adjusting the interaction between variables to make the network generate observed data</p> <p>Can be trained without supervision</p> <p>Training using very fast algorithm to initialize the network and parameters</p> <p>Can quickly get unbiased sample from posterior distribution</p>	<p>It is limited to binary inputs</p> <p>The parameters are fine-tuned using slow algorithm</p>	<p>Image recognition</p> <p>Face recognition</p> <p>Video sequence recognition</p> <p>Motion-capture data</p>	<p>(Wang et al. 2018)</p> <p>(Chandrayan 2017)</p>
Restricted Boltzmann Machine (RBM)	<p>Automatic training for the datasets avoiding the local minimum values</p> <p>Robust to ambiguous inputs</p>	<p>It is limited to binary inputs</p> <p>There is no output layer</p> <p>Restricted connection between hidden units</p>	<p>Filtering</p> <p>Prediction</p>	<p>(Wang et al. 2018)</p> <p>(Chandrayan 2017)</p>

(Continued)

Table 1. (Continued).

Deep Learning Network	Advantages	Limitations	Applications	References
Recurrent Network (RN)	It has the ability to make structured predictions	Used only when applying same weights recursively	Speech recognition	(Davydova 2017)
Auto Encoder Network (AEN)	When it is trained, the output is as close to input as possible Can be trained without supervision Eliminating the irrelevance in the input and preserve the meaningful information	Used only when output cells are equal to input cells and hidden cells is less than input cells Error propagation	Classification Clustering Feature compression	(Wang et al. 2018) (Tch 2017)
Generative Adversarial Network (GAN)	Very large family of double networks Learning a map from an input to output image	Generator and discriminator fool each other	Image generation	(Tch 2017)

Convolutional Neural Network

This section presents a brief review of CNN principles in order to find gaps and open issues that need to be solved in order to improve the performance of CNN created models.

Definition and Way of Working

CNN is one of the most popular DL networks and is used mainly to perform image classification tasks. (MathWorks-1) stated that it is useful for detecting patterns in images that help in the automatic recognition of real physical objects. These patterns are extracted by CNN's directly from image datasets without the need for extracting feature manually, which is the most important factor the makes CNN very popular. In addition, it produces highly accurate recognition results and has the flexibility to be retrained to perform new recognition and to be built on previously created models. It provides an optimal model architecture, enabling advances in detecting and recognizing objects, thus it is a key technology in automated facial recognition. CNN might have hundreds of layers helping to detect patterns in images, filters are used to extract information and can start from simple features, such as brightness, to more complex ones that uniquely identify an object. This filtering is applied to each training image and the output of each image after convolution can be used as an input to the following layer. CNNs are composed of an input layer, an output layers, and many hidden layers. Every neuron in the hidden layer connects to all inputs neurons as shown in [Figure 1](#), as mentioned by Le (2015):

The hidden layers perform learning feature tasks using three most common feature learning layers which are:

- Convolution: it activates some features in the images using convolutional filters that represented by matrix of weights that slide along the pixel brightness input matrix to created feature map matrix using special dot product as mentioned in by Hui (2017).

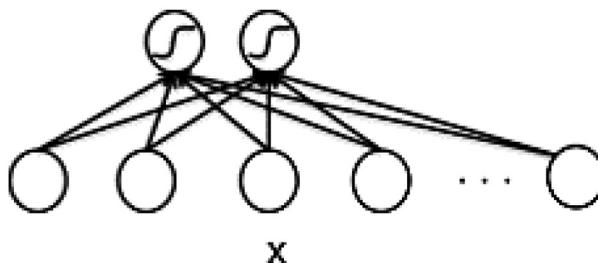


Figure 1. Inputs and hidden layers connection in CNN (Le 2015).

- Rectified linear unit (ReLU): it is used after each convolutional layer to increase the speed and effectiveness of training by mapping negative values to zeros and maintaining positive values, which is helpful as an activation.
- Batch normalization: it is used as supplement layers after each convolutional layer to mitigate the risk of overfitting by normalizing the input values of the following layers (Yamashita et al. 2018).
- Pooling: it is used between the convolutional layers to reduce the dimensionality of the output volume (McDermott 2021) without losing the important features that contribute to minimize the computational cost, it reduces the number of parameters needed to learn by making nonlinear down-sampling that simplifies the output. There are two types max pooling that take the most activated feature and average pooling that take the average presence of the feature, so max pooling is better with dark background and average pooling is better with white background as mentioned in Ouf (2017).

Furthermore, it uses two classification layers:

- Fully connected: it shows the probability of each image being classified for each class.
- SoftMax: it is an activation function that works better with multi-class classification problems rather binary classification problem that requires sigmoid logistic function as shown in McDermott (2021), it provides the classification output and may not have any parameters as mentioned by Wu (2017).

Figure 2 below demonstrates the way of working for CNN showing feature learning and classification layers:

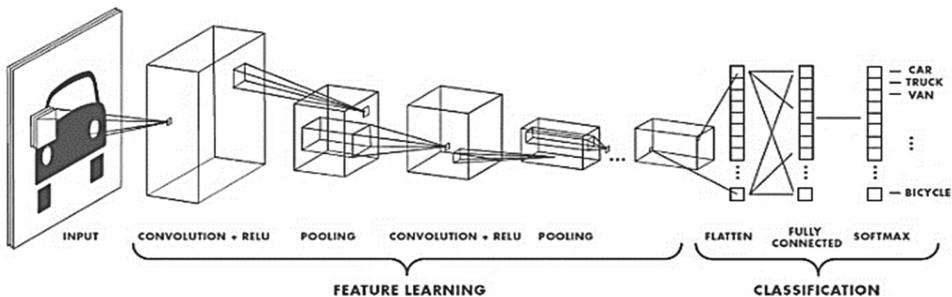


Figure 2. The way of working for CNN (MathWorks-1).

Gaps and Open Issues

According to Joshi et al. (2019), the most important challenge in training CNN is the generalization to unseen datasets so that the model does not overfit data sets and can give judgment on unknown data. Overfitting is a common issue in training CNN, where the model fits well enough to the training data but does not have the capability to generalize to other datasets. It can be controlled by increasing the sample data using data augmentation techniques, reducing the complexity of the architecture and stopping the training earlier. In addition, Liang and Liu (2015) and Cogswell et al. (2015) discussed the same overfitting issue and suggested another way to prevent it using dropout, Pan (2017) highlighted that it is an efficient method to randomly remove a unit from the network with related edges independently for each hidden unit and sample. Wu (2017) used novel regularization technique that helps in reducing kernel redundancy and thus preventing overfitting. However, improving the learning capability is still an open challenge and can be enhanced continuously. In addition, Joshi et al. (2019) reported other issues, such as exploding gradient problem where the model stops learning after certain number of epochs which causes instability in learning process resulting in NAN values, this issue can be overcome by redesigning the network architecture and selecting appropriate activation function. Also, Shah et al. (2016) suggested using residual and highway networks that learn in earlier layers allowing for earlier representation. In addition to Joshi et al. (2019), Fu et al. (2016) and Masko and Hensman (2015) addressed the third issue which is training the data using imbalanced classes where the sample is not uniformly distributed, it is a significant and long-standing challenge in training CNN models. Improving the convergence speed is another future challenge because it sometimes increases the time of convergence in order to get better accuracy as stated by Chiroma et al. (2019).

Furthermore, five research papers Zhang, Kiranyaz, and Gabbouj (2018), Baldominos, Saez, and Isasi (2018), Sinha, Verma, and Haidar (2017), Ma et al. (2018) and Panwar et al. (2017) discussed the most challenging aspect in training CNN network, which is designing better topology, the traditional heuristic approach of using trial and error might result in less accurate model depending on user experience. Applying optimization techniques such as nature inspired algorithms to optimize the parameters of CNN network can improve the performance of the model. However, designing better CNN topology is still an open issue and there is no approach found yet that can give the best CNN topology. The following section will discuss the impact of developing hybrid CNN network in increasing the accuracy results.

Nature Inspired Algorithms in Deep Learning

This section will show the impact of integration nature inspired algorithms with DL networks and proposes a novel hybrid algorithm.

Impact and Recent Applications

Chiroma et al. (2019) discussed the synergy between nature inspired algorithms with DL. They mentioned that the inspiration of such algorithms can be from animals' behaviors, human activities, and biological systems. The paper presented many nature inspired algorithms such as harmony search, firefly, cuckoo search, evolutionary, ant colony optimization, practical swarm optimization, genetic, simulated annealing, and gravitational search algorithm. The authors stated that combining DL with nature inspired algorithms has the advantage of solving local minima problem and improving the performance of the network by increasing the accuracy of its architecture. In addition, the need for trial-and-error techniques in determining the parameters of DL architecture is eliminated as nature inspired algorithms realize the best parameters values automatically. Although the optimum parameters setting is still an open problem in the research area. The authors suggested to eliminate the need for human interventions in determining the parameters by obtaining parameter-less nature inspired algorithms in the future. Finally, the paper suggested to apply meta-optimization that is excessive in the DL area, and it helps to tune optimization methods by using another optimization method.

Furthermore, other research papers discussed the hybrid CNN with nature inspired sward-based optimization techniques such as CNN with Evolutionary Algorithm (EA-CNN) and CNN with Genetic Algorithm (GA-CNN) in addition to other techniques like CNN with Long Short-Term Memory (LSTM-CNN) and CNN with Fuzzy Logic (FL-CNN). Table 2 summarize the accuracy of original CNN and improved accuracy after hybridization.

Table 2. Hybrid CNN accuracy.

Hybrid Algorithm	Accuracy		Dataset	Reference
	Original CNN	Hybrid CNN		
(EA-CNN)	-	98.88%	MNIST database for handwritten digits recognition	(Badan 2019)
	-	62.37%	CIFAR10 dataset for animals' image classification	
(GA-CNN)	71.69%	75.95%	Stock market fluctuation prediction	(Chung & Shin, 2019)
(LSTM-CNN)	82.1%	97%	Motion data and site video to recognize workers unsafe actions	(Ding et al. 2018)
(FL-CNN)	97.35%	99.10%	Handwritten digits recognition	(Popko and Weinstein 2016)

Applying evolutionary algorithm yield a highly accurate CNN with 98.88% accuracy for handwritten digit recognition and lower percentage of 62.37% for animal image classification, the author used weight inheritance technique which consider the training process as kind of mutation that reduce the evolution cycle time. Bernard and Leprévost (2018) explained the process of evolution by reproducing the population through generation by crossing members and inducing random mutation, evolving input image would maximize feature activation. Genetic algorithm is one of the evolutionary algorithms that has been applied to optimize the parameters of CNN model to predict the stock market, Mallawaarachchi (2017) described the natural selection of selecting the fittest individual in the population which helped improve the accuracy from 71.69% to 75.95%, the algorithm produces offspring that inherit the parents' characteristics, so that they have a chance to survive if their parents have a better fitness, so the algorithm consists of five phases: initial population, fitness function, selection, crossover, and mutation.

In addition to nature inspired algorithms, CNN can be integrated with other DL algorithm called long-short-term memory that automatically recognize worker unsafe actions in motion data, the use of LSTM would enable the sequence of learning features, dealing with sequential data is an important advantage of using this algorithm as stated by Motepe, Hasan, and Stopforth (2019), it is an effective technique when capturing dependencies in the long-term avoiding recurrent neural network challenges such as vanishing gradient problem using nonlinear gating that regulate the flow of information. Furthermore, the hybridization of CNN with FL would add one more layer, a fuzzy self-organization layer, Korshunova (2018) explained its function that distribute input data into clusters do not equivalent to number of output classes where the output of this layer is the membership functions values for the fuzzy clusters. This new hybrid model improved the handwritten digits recognition accuracy from 97.35% to 99.10%.

However, there are still opportunities to test the integration, with CNN, of other popular swarm-based algorithms, such as artificial bee colony. Such integration was done in other applications, for example to optimize the hyperparameters of ANN as presented by (Rashid and Abdullah (2018). They integrated artificial bee colony, genetic algorithm, and back propagation neural network to classify and diagnose diabetes. Adding genetic operator helps avoid sucking in local optima, an issue mentioned by Packianather et al. (2014). In addition, Bullinaria and AlYahya (2014) made a comparison between training ANN with back propagation and BA, and they found that back propagation is significantly better than artificial bee colony. Xu et al. (2019) applied a modified artificial bee colony that have better performance in utilizing the neighbor information in order to accelerate the convergence, this new algorithm was used to train ANN. Qolomany et al. (2017) optimized two variables of DL model, which are number of hidden layers and number of neurons in each layer using particle swarm optimization. Furthermore, Badem et al. (2017) applied BA along with limited memory Broyden–Fletcher–Goldfarb–Shannon

to train autoencoder network while Lee, Park, and Sim (2018) optimized the hyperparameters of CNN using free harmony search technique. Looking at the literature, the hybridization between bees algorithm and CNN is an important gap which will be addressed in the following section by proposing novel hybrid algorithm that takes the advantage of BA to train CNN.

In addition, nature inspired algorithm can be hybridized with other deep-learning networks such as deep Recurrent Neural Network (RNN), (Zeybek et al. 2021) presented novel metaheuristic algorithm that train deep RNN using an enhanced ternary bee's algorithm (BA-3+) for sentiment classification task. BA-3+ algorithm finds the optimal set of parameters for deep RNN architecture by collaborative search of three bees, the authors found that it outperformed other optimization algorithms such as stochastic gradient descent (SGD), differential evolution (DE) and particle swarm optimization (PSO). Training deep RNN using BA-3+ algorithm achieved accuracy rate between 80%-90% while training it using SGD produced accuracy between 50% and 60% for most datasets.

Proposed Novel Hybrid Algorithm

A new approach for optimizing the parameters and weights of CNN through BO and BA will be explained. First, design of experiments technique will be conducted to investigate the significant factors affecting validation accuracy (VA). Then, BO will be used to find the optimal hyperparameters for the network by optimizing the significant factors in order to minimize the classification error on the validation set, which is the objective function. (MathWorks-2) stated the benefits of the parameters and defined a specific range for each of them as shown in Table 3.

Table 3. CNN parameters information.

Variable	Benefit	Range
Section Depth (SD)	Control the depth of the network	1-3
Initial Learning Rate (ILR)	Allow for learning the features	$1e^{-2} - 1$
Momentum (M)	Update the hyperparameters	0.8-0.98
Regularization (R)	Prevent overfitting	$1e^{-10} - 1e^{-2}$

Four factors and three levels will be used in the experiments as shown in following table:

Table 4. Factors and levels definition.

Factors	Level 1	Level 2	Level 3
Section Depth	1	2	3
Initial Learning Rate	0.01	0.055	0.1
Momentum	0.88	0.9	0.92
Regularization	$1e^{-10}$	$2.5e^{-4}$	0.0005

Taguchi orthogonal array (L9) is designed for nine experiments of applying original CNN with 9 different combinations of parameters on ‘Cifar10DataDir’ dataset which is benchmark image data that consists of 10 classes airplane, automobile, bird, cat deer, dog, frog, horse, ship, and truck. Each class has 6,000 images so, the total sample size is 60,000 images, the results of VA are shown in the following table:

Table 5. Taguchi orthogonal array (L9).

Exp #	SD	ILR	M	R	VA (%)
1	1	0.01	0.88	1e-10	67.96
2	1	0.055	0.9	2.5e-4	72.20
3	1	0.1	0.92	0.0005	70.90
4	2	0.01	0.9	0.0005	76.40
5	2	0.055	0.92	1e-10	78.44
6	2	0.1	0.88	2.5e-4	77.40
7	3	0.01	0.92	2.5e-4	77.74
8	3	0.055	0.88	0.0005	79.56
9	3	0.1	0.9	1e-10	80.34

After Running the experiments and recording VA for each run, analysis of variance (ANOVA) has been applied using Minitab software in order to investigate the most influential factors affecting VA as shown below:

Factor Information

Factor	Type	Levels	Values
Section Depth	Fixed	3	1, 2, 3
Initial Learning Rate	Fixed	3	0.010, 0.055, 0.100
Momentum	Fixed	3	0.88, 0.90, 0.92
Regularization	Fixed	3	0.0000000, 0.0002500, 0.0005000

Analysis of Variance

Source	DF	Adj SS	Adj MS	F-Value	P-Value
Section Depth	2	131.583	65.7916	*	*
Initial Learning Rate	2	12.313	6.1564	*	*
Momentum	2	2.698	1.3492	*	*
Regularization	2	0.067	0.0336	*	*
Error	0	*	*		
Total	8	146.662			

Model Summary

S	R-sq	R-sq(adj)	R-sq(pred)
*	100.00%	*	*

Figure 3. ANOVA Results.

Looking at the results, it is revealed that all four factors have significant effect on VA, so all of them will be optimized using BO technique to minimize the classification error on the validation set, which is the objective function, it is a function of section depth, initial learning rate, momentum, and regularization:

Minimize Classification Error on Validation Set = CNN Training and Validation (SD, ILR, M, R)

BO algorithm builds the probability model of the objective function to be used to select the hyperparameters of the network and then evaluate them in the true objective function, it maintains a Gaussian process model in the objective function that uses network variables as inputs to specify the network architecture. It is applied in conjunction with stochastic gradient descent momentum (SGDM) as one of the training options in CNN architecture, momentum adds inertia that helps the current update to make proportional contribution to the previous iteration update.

BA is used to optimize the weight learning rate factor to adjust the global learning rate obtained by BO algorithm in each of the three convolutional layers and fully connected layer. Optimal learning rate means the optimum amount of weight update (Brownlee 2020) in the convolutional filters and fully connected layer that perform the classification, so the classification accuracy on validation set is improved.

According to Al-Musawi (2019) BA is one of the most important swarm-based optimization techniques that perform an intense search after local search to optimize the variables, so combining it with BO technique will contribute to optimize CNN parameters that minimize the classification error on validation set. Its way of working is inspired by honeybees foraging behavior. The algorithm requires to set the following parameters:

- Number of scout bees: n
- Number of selected bees: m
- Number of elite bees: e
- Number of recruited bees for elite (e) sites: nep
- Number of recruited bees for other best ($m-e$) sites: nsp
- Neighborhood size for each selected patch (local search): ngh

The algorithm starts with the global search when the scout bees (n) arrive at random positions and evaluate them based on the fitness value, then a local search stage starts by selecting the best sites (m) and abandoning the remaining sites. After that an intense search is initiated by selecting elite sites (e), which are the best among the best sites. The next step is selecting the size of the neighborhood search space in order

to recruit more bees for the elite sites (e) and fewer bees for non-elite sites (m-e) to conduct local search. The global and local search will be performed simultaneously, while the recruited bees are exploring the best solutions around the neighborhood, the global search on the remaining sites is carried out randomly. This iterative process is stopped when the optimal solution found, the iteration number exceeded or no improvement over specific sequential number of iterations.

In addition to the basic Baronti (2020) showed other two types of this technique shrinking and standard algorithm, the idea behind the shrinking approach is taking the samples from increasingly small regions in solution space during the local search while the standard algorithm includes abandoning a site when stagnating local search in addition to shrinking procedure. Lindfield and Penny (2017) introduced further modification by counting the number of times a recruited bee failed to explore an improved site, it is used as a guide in the exploration process to improve the efficiency and effectiveness of the search. Other enhancements that were introduced by Imanguliyev (2013) include an early neighborhood efficient algorithm, hybrid Tabu bee's algorithm and autonomous bees' algorithm. Packianather, Al-Musawi, and Anayi (2019) proposed a new version of BA discovering the rule automatically by adding two parameters namely quality weight and coverage weight to avoid ambiguous situations in the prediction stage. They formulated the new two parameters to carry out meta pruning and make the algorithm suitable for classification tasks, it achieved better classification accuracy and reduced the number of rules making it more efficient algorithm than other classification methods such as Jrip and other evolutionary algorithms.

The novel hybrid Bees Bayesian CNN (BA-BO-CNN) uses the basic BA to optimize the weight learning rate factor to adjust the global learning rate obtained by BO algorithm in each of the three convolutional layers and fully connected layer. Applying BA is after optimizing section depth, initial learning rate, momentum, and regularization parameters using BO method. The values of hyperparameters for bee's algorithm is assigned based on the computer capability and equations in (MathWorks-3):

- Maximum number of iterations = 1
- Number of scout bees (n) = 6
- Number of selected bees (m) = $0.5 * n = 0.5 * 6 = 3$
- Number of elite bees (e) = 1
- Number of recruited bees for elite (e) sites (nep) = $2 * m = 2 * 3 = 6$

- Number of recruited bees for other best (m-e) sites (nsp) = $0.5 * n = 0.5 * 6 = 3$
- Neighborhood size for each selected patch (local search) (ngh) = $0.1 * (\text{Var max} - \text{Var min}) = 0.1 * (1.1 - 0.9) = 0.1 * 0.2 = 0.02$

The pseudocode for the proposed hybrid BA-BO-CNN algorithm is shown in [Figure 4](#):

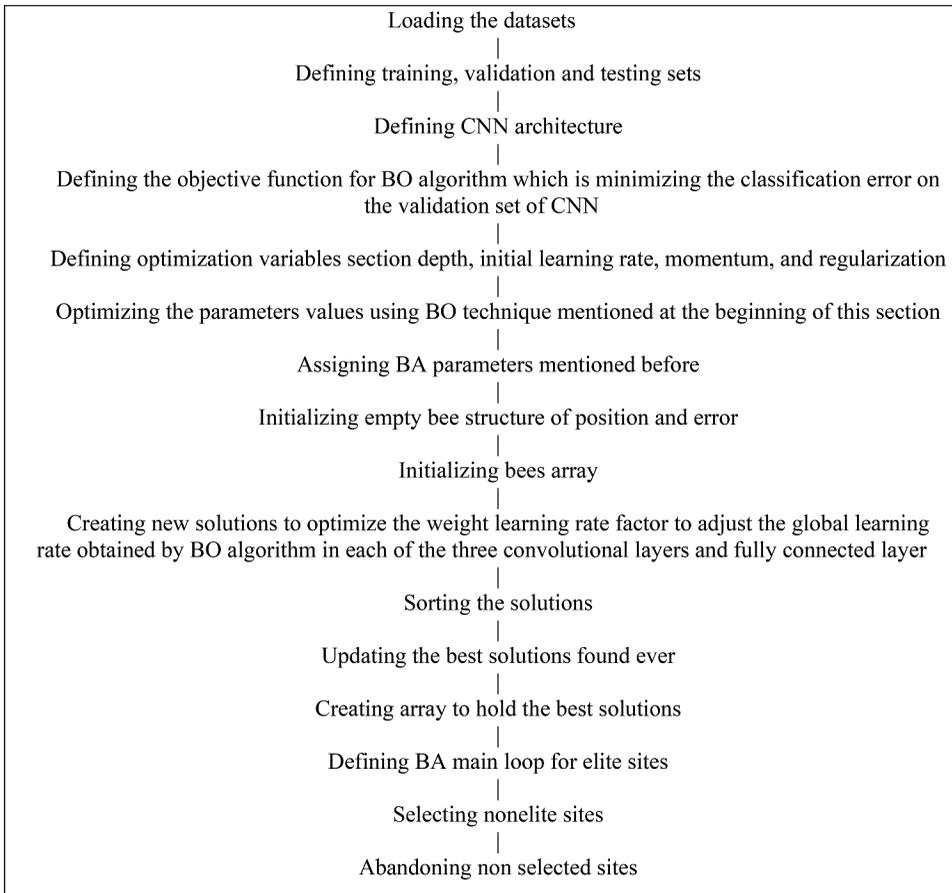
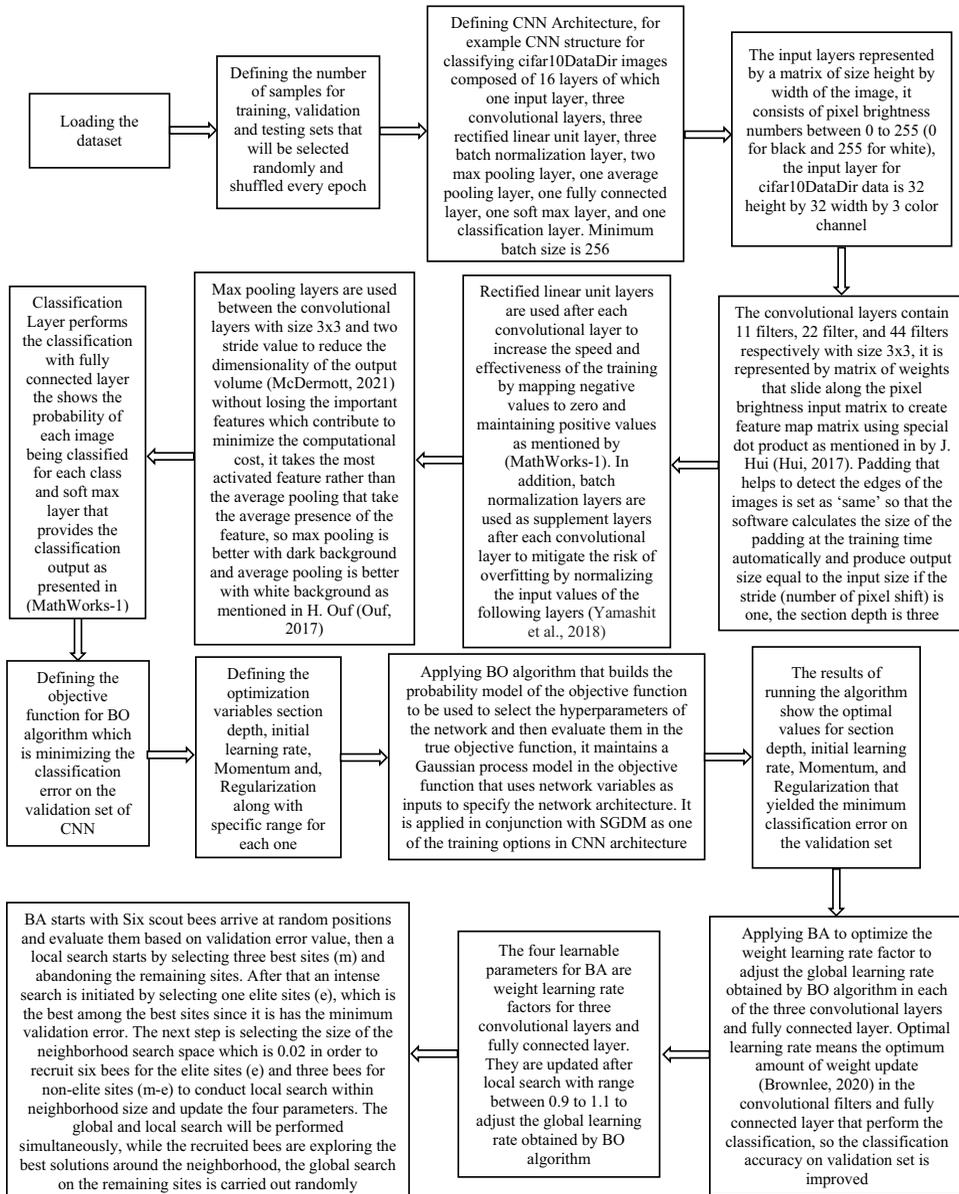


Figure 4. The pseudo code for BA-BO-CNN.

In addition to pseudocode, the following flow chart explains CNN architecture along with BO and BA showing how they perform to optimize CNN parameters:



Results and Discussion of Proposed BA-BO-CNN

MATLAB software is used to apply the hybrid BA-BO-CNN algorithm on three benchmark datasets, cifar10DataDir benchmark image data that consists of 60,000 images classified evenly into 10 classes airplane, automobile, bird, cat, deer, dog, frog, horse, ship, and truck. The second dataset is handwritten digits with ten classes from 0 to 9 with 1,000 images in each class. The last set of data is concrete crack images with two classes, 5,000 negative images without cracks present in the road and 5,000 positive images with cracks.

First, BO technique is applied on cifar10DataDir benchmark image data to find the optimum CNN parameters, Figure 5 shows the minimum observed objective function and estimated minimum objective function:

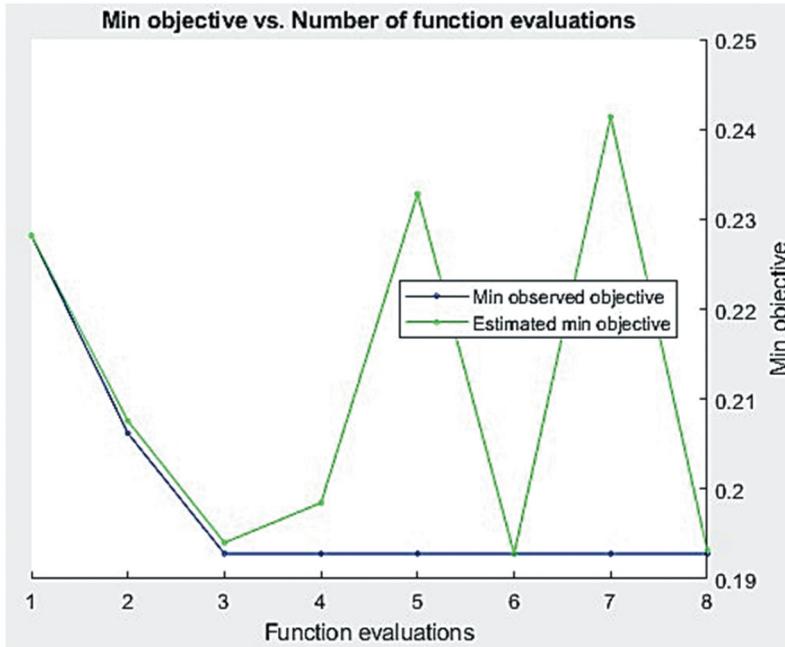


Figure 5. Number of functions evaluations.

Figure 6 shows the optimum parameters values for section depth, initial learning rate, momentum, and regularization that yielded the minimum classification error obtained by BO technique:

```

Optimization completed.
MaxTime of 14400 seconds reached.
Total function evaluations: 8
Total elapsed time: 14670.6496 seconds
Total objective function evaluation time: 14660.1188

Best observed feasible point:
  SectionDepth  InitialLearnRate  Momentum  L2Regularization
-----
          3          0.68934      0.84074      4.5857e-05

Observed objective function value = 0.1928
Estimated objective function value = 0.19318
Function evaluation time = 2650.7429

Best estimated feasible point (according to models):
  SectionDepth  InitialLearnRate  Momentum  L2Regularization
-----
          3          0.68934      0.84074      4.5857e-05

Estimated objective function value = 0.19318
Estimated function evaluation time = 3081.7537

```

Figure 6. Optimal values for CNN parameters.

The minimum classification error value is 0.1928, it is a result of section depth value of 3, initial learning rate of 0.68934, momentum of 0.84074 and regularization of $4.5857e-05$.

Then, BA is added to adjust the global learning rate in each of the three convolutional layers and fully connected layer, [Figure 7](#) shows the optimal weight learning rate factors optimized by BA along with classification error on validation set:

Field ▲	Value
Position	[0.9308,1.0924,1.0753,0.9977]
Error	0.1778

Figure 7. Optimal weight learning rate factors.

So, the global learning rate of 0.68934 obtained by BO is adjusted by multiplying it by 0.9308 in the first convolutional layer to become 0.6416. In the section convolutional layer, the adjustment factor is 1.0924 resulting in a learning rate value of 0.7530, the factor for third convolutional layer is 1.0753, so the adjusted value is 0.7412. Finally, the new learning rate value for fully connected layer is 0.6877 after adjusting it by a factor of 0.9977. [Figure 8](#) shows the training progress and validation accuracy after adding BA, the training progress curve for CNN, BO-CNN and BA-BO-CNN are similar:

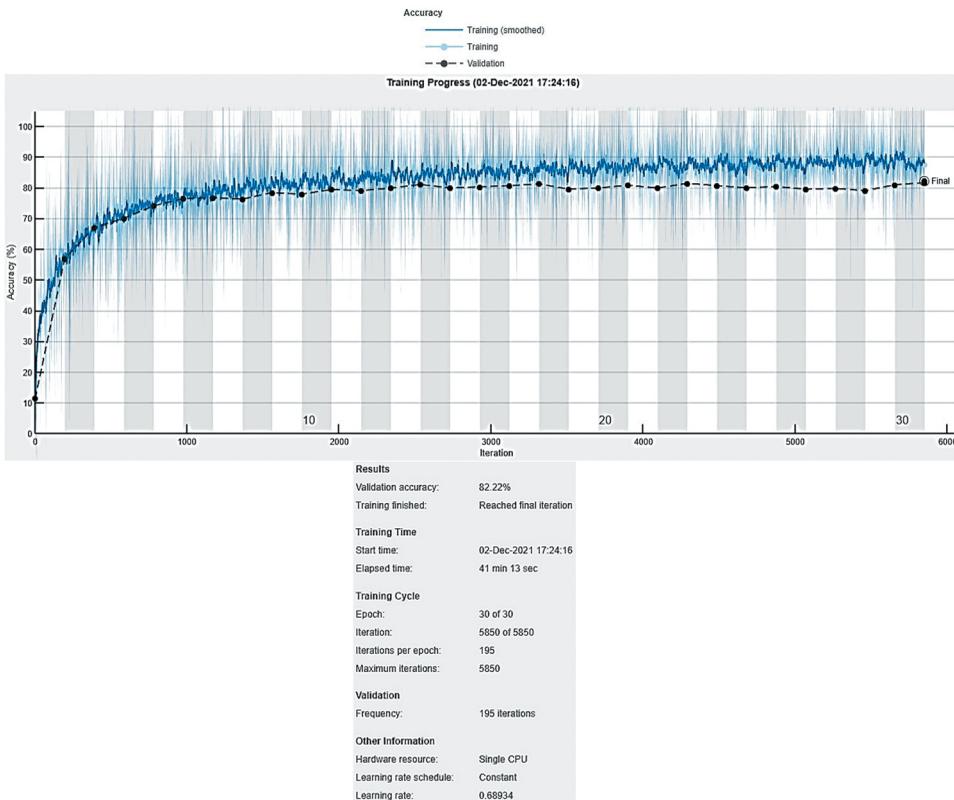


Figure 8. Training progress for BA-BO-CNN.

The following three figures present the confusion matrix for training, validation and testing set for BA-BO-CNN showing precision and false alarm in the blue and red rows, and recall and miss out in the blue and red columns respectively:

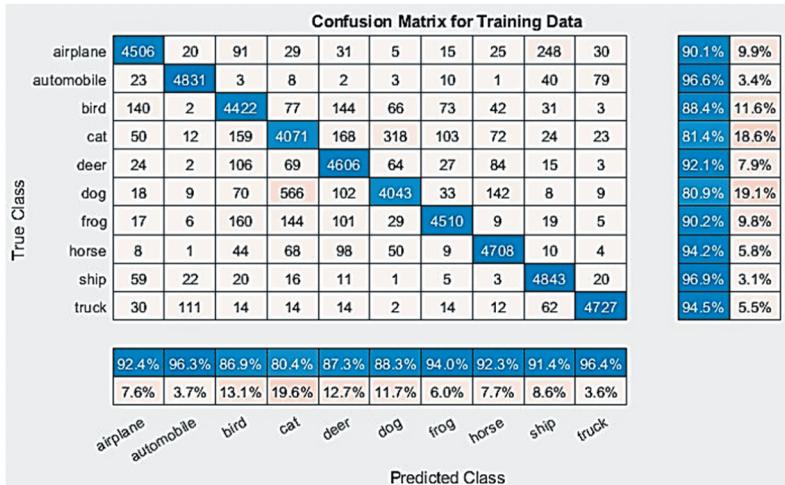


Figure 9. Training confusion matrix for BA-BO-CNN.

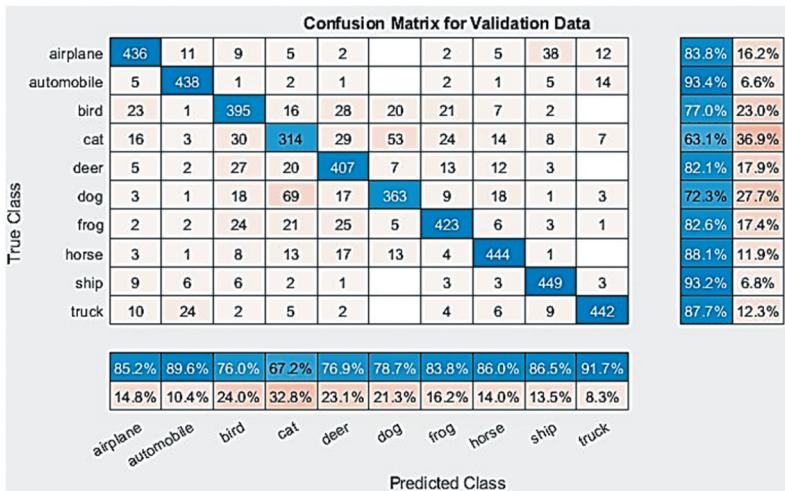


Figure 10. Validation confusion matrix for BA-BO-CNN.

Confusion Matrix for Testing Data

True Class	Confusion Matrix for Testing Data										Precision	
	airplane	automobile	bird	cat	deer	dog	frog	horse	ship	truck	airplane	automobile
airplane	384	4	22	9	6	1	5	1	40	8	80.0%	20.0%
automobile	6	483	3	2			1	1	8	27	91.0%	9.0%
bird	30	2	338	22	37	19	21	12	6		69.4%	30.6%
cat	12	3	30	330	31	51	20	13	9	3	65.7%	34.3%
deer	5	1	24	20	416	7	6	16	8	1	82.5%	17.5%
dog	2	2	9	78	15	352	8	23	7	2	70.7%	29.3%
frog	4		34	20	18	3	404	2	2	1	82.8%	17.2%
horse	8		14	16	18	10	2	424	3	1	85.5%	14.5%
ship	13	10	5	5	3	2	1		472	7	91.1%	8.9%
truck	11	25	4	1				2	6	7	88.7%	11.3%

Precision		Recall		F1 Score		Accuracy		Specificity	
airplane	automobile	bird	cat	deer	dog	frog	horse	ship	truck
80.8%	91.1%	70.0%	65.6%	76.5%	79.1%	86.0%	85.1%	84.0%	89.8%
19.2%	8.9%	30.0%	34.4%	23.5%	20.9%	14.0%	14.9%	16.0%	10.2%

Predicted Class

Figure 11. Testing confusion matrix for BA-BO-CNN.

The following table shows the training, validation, and testing accuracy along with computational time for original CNN, BO-CNN, and BA-BO-CNN:

Table 6. Classification accuracy and computational time for algorithms (cifar10DataDir dataset).

	Existing Original CNN	Existing BO-CNN	Novel Hybrid BA-BO-CNN
Training Accuracy	92.68%	90.27%	90.53%
Validation Accuracy	80.34%	80.72%	82.22%
Testing Accuracy	80.54%	80.69%	80.74%
Computational Time	42 Min 56 Sec	43 Min 24 Sec	41 Min 13 Sec

It is seen that the best training accuracy is 92.68% for the original CNN, while the best validation and testing accuracy are 82.22% and 80.74% achieved by novel Hybrid BA-BO-CNN. The improved validation accuracy from BO-CNN to BA-BO-CNN is 1.5%, which is higher than the improved accuracy of 0.38% between existing original CNN and existing BO-CNN. Also, it has better performance than EA-CNN model designed by F. Badan (Badan 2019) that achieved an accuracy of 62.37% on the same cifar10DataDir dataset as was shown in Table 2. So, it is concluded that novel hybrid BA-BO-CNN has better classification performance and generalization capability to unseen data. In addition, it is the best algorithm in terms of cost-effectiveness since it achieved lower computational time than BO-CNN by 2 min and 11 s.

The same procedure is followed to apply original CNN, BO-CNN, and novel hybrid BA-BO-CNN on other two datasets, the results are shown in the following two tables:

Table 7. Classification accuracy and computational time for algorithms (digits dataset).

	Existing Original CNN	Existing BO-CNN	Novel Hybrid BA-BO-CNN
Training Accuracy	99.94%	100%	100%
Validation Accuracy	98.80%	100%	100%
Testing Accuracy	99.20%	99.99%	99.99%
Computational Time	2 Min 35 Sec	6 Min 42 Sec	3 Min 30 Sec

Table 8. Classification Accuracy and computational time for algorithms (concrete cracks dataset).

	Existing Original CNN	Existing BO-CNN	Novel Hybrid BA-BO-CNN
Training Accuracy	99.96%	99.90%	99.95%
Validation Accuracy	99.50%	99.67%	99.67%
Testing Accuracy	99.83%	99.30%	99.35%
Computational Time	3 Min 20 Sec	2 Min 46 Sec	2 Min 50 Sec

The table shows that the novel hybrid BA-BO-CNN produces exactly the same accuracy due to simple features in the digits images which can be classified with high accuracy using existing algorithms, but the computational time in the hybrid algorithm is better by 3 minutes and 12 seconds reduction, so it is better in terms of cost effectiveness.

In this dataset, the new hybrid algorithm has almost similar results to existing BO-CNN since it has only two classes, so applying the existing algorithms produced high accuracy.

Conclusion

Artificial Neural Network has the ability to handle high-dimensional real-time data and extract implicit meaningful patterns that can be used to predict the outcome for previously unseen cases. The performance of ANNs depend on the significance of the features extracted from the training data used in training which is often a time-consuming process leading to sub-optimal solutions. However, this problem could be overcome by Deep Learning (DL), which is a type of Machine Learning technique. DL has better learning capability as it has the advantage of automatic feature extraction by learning large number of nonlinear filters before making decisions.

In this paper, a review on DL has been conducted showing all its networks along with their advantages, limitations, and applications. The most popular DL network is CNN that use four feature learning layers, namely, convolution, rectified linear unit batch normalization and pooling layers in addition to two classification layers one of which is fully connected and the other is a SoftMax layer used mainly for image classification. It has been found from the review

that there are five open issues namely, overfitting, exploding gradient problem, training the model using imbalanced classes, the convergence speed and designing better CNN topology which is the most important issue that need to be addressed in order to improve the performance of CNN models further. In addition, nature inspired algorithms have been included showing its contribution to increase the classification accuracy.

This paper addressed designing better CNN topology issue by proposing a novel nature inspired hybrid algorithm that combines Bayesian Optimization (BO) with Bees Algorithm (BA) in order to increase the overall performance of CNN which is referred to as BA-BO-CNN. BO optimized four parameters section depth, initial learning rate, Momentum, and Regularization while BA optimized the weight learning rate factor to adjust the global learning rate obtained by BO algorithm in each of the three convolutional layers and fully connected layer.

Applying the hybrid algorithm on Cifar10DataDir benchmark image data yielded an increase in the validation accuracy from 80.72% to 82.22%, while applying it on digits datasets showed the same accuracy as the existing original CNN and BO-CNN, but with an improvement in the computational time by 3 min and 12 s reduction, finally applying it on concrete cracks images produced almost similar results to existing algorithms.

In the future, the novel hybrid BA-BO-CNN algorithm can be developed further by optimizing the weight regularization factor in the convolutional layers and fully connected layer using BA in order to improve the performance of CNN further.

Acknowledgment

I would like to thank the Saudi Arabian Cultural Bureau and Cardiff University for their support.

Disclosure Statement

No potential conflict of interest was reported by the author(s).

Funding

This work was supported by the Saudi Arabia Cultural Bureau in London

ORCID

Nawaf Mohammad H Alamri  <http://orcid.org/0000-0002-5641-0178>

Michael Packianather  <http://orcid.org/0000-0002-9436-8206>

Samuel Bigot  <http://orcid.org/0000-0002-0789-4727>

References

- Al-Musawi, A. 2019. The development of new artificial intelligence based hybrid techniques combining bees algorithm, data mining and genetic algorithm for detection, classification and prediction of faults in induction motors. Doctoral dissertation, Cardiff University.
- Badan, F. 2019. Evolutionary algorithms in convolutional neural network design. Online Accessed March 16, 2020. <http://excel.fit.vutbr.cz/submissions/2019/033/33.pdf>
- Badem, H., A. Basturk, A. Caliskan, and M. E. Yuksel. 2017. A new efficient training strategy for deep neural networks by hybridization of artificial bee colony and limited-memory BFGS optimization algorithms. *Neurocomputing* 266:506–26. doi:10.1016/j.neucom.2017.05.061.
- Baldominos, A., Y. Saez, and P. Isasi. 2018. Evolutionary convolutional neural networks: An application to handwriting recognition. *Neurocomputing* 283:38–52. doi:10.1016/j.neucom.2017.12.049.
- Baronti, L. (2020). Analysis and development of the bees algorithm for primitive fitting in point cloud models. Doctoral dissertation, University of Birmingham.
- Bernard, N., and F. Leprévost. 2018. Evolutionary algorithms for convolutional neural network visualisation. In Latin American High Performance Computing Conference, Springer, Cham, September.
- Brownlee, J. (2020). Understand the impact of learning rate on neural network performance. Online [Accessed November 29, 2021. <https://machinelearningmastery.com/understand-the-dynamics-of-learning-rate-on-deep-learning-neural-networks/>
- Bullinaria, J. A., and K. AlYahya. 2014. Artificial bee colony training of neural networks: Comparison with back-propagation. *Memetic Computing* 6 (3):171–82. doi:10.1007/s12293-014-0137-7.
- Chandrayan, P. 2017. Deep learning: deep belief network fundamentals. Online Accessed October 8, 2019. <https://codeburst.io/deep-learning-deep-belief-network-fundamentals-d0dcfd80d7d4>
- Chiroma, H., A. Y. U. Gital, N. Rana, M. A. Shafi'i, A. N. Muhammad, A. Y. Umar, and A. I. Abubakar. 2019. Nature inspired meta-heuristic algorithms for deep learning: recent progress and novel perspective. In Science and Information Conference. Springer, Cham, April.
- Chung, H., & Shin, K. S. 2019. Genetic algorithm-optimized multi-channel convolutional neural network for stock market prediction. *Neural Computing and Applications* 1–18.
- Ciortea, E. M. 2018. IoT analysis of manufacturing using petri nets. In IOP Conference Series: Materials Science and Engineering Vol. 400, No. 4, IOP Publishing, Constanta, Romania, August.
- Cogswell, M., F. Ahmed, R. Girshick, L. Zitnick, and D. Batra. 2015. Reducing overfitting in deep networks by decorrelating representations. arXiv preprint arXiv:1511.06068..
- Davydova, O. 2017. 7 types of artificial neural networks for natural language. Online Accessed October 8, 2019. <https://medium.com/@datamonsters/artificial-neural-networks-for-natural-language-processing-part-1-64ca9ebfa3b2>
- De Filippis, L. A. C., L. M. Serio, F. Facchini, and G. Mummolo. 2017. ANN Modelling to Optimize Manufacturing Process. In *Advanced Applications for Artificial Neural Networks*, ed. Adel El-Shahat, (pp. 201–226). IntechOpen.
- Ding, L., W. Fang, H. Luo, P. E. Love, B. Zhong, and X. Ouyang. 2018. A deep hybrid learning model to detect unsafe behavior: Integrating convolution neural networks and long short-term memory. *Automation in Construction* 86:118–24.
- Ebubekir, K. 2010. *The bees algorithm theory, improvements and applications*. Cardiff United Kingdom.: Manufacturing Engineering Centre School of Engineering University of Wales.

- Ekins, S. 2016. The next era: Deep learning in pharmaceutical research. *Pharmaceutical Research* 33 (11):2594–603. doi:10.1007/s11095-016-2029-7.
- Fu, K., D. Cheng, Y. Tu, and L. Zhang. 2016. Credit card fraud detection using convolutional neural networks. In *International Conference on Neural Information Processing*, Springer, Cham, October.
- Geissbauer, R., J. Vedso, and S. Schrauf. 2016. Global industry 4.0 survey: Building the digital enterprise, 1. Retrieved from PwC Website: <https://www.pwc.com/gx/en/industries/industries-4.0/landing-page/industry-4.0-building-your-digital-enterprise-april-2016.pdf>
- Hui, J. 2017. Convolutional neural networks (CNN) tutorial. Online Accessed November 15, 2021. <https://jhui.github.io/2017/03/16/CNN-Convolutional-neural-network>
- Imanguliyev, A. 2013. Enhancements for the Bees Algorithm. Doctoral dissertation, Cardiff University.
- Joshi, S., D. K. Verma, G. Saxena, and A. Paraye. 2019. Issues in training a convolutional neural network model for image classification. In *International Conference on Advances in Computing and Data Sciences*, Springer, Singapore, April.
- Korshunova, K. P. 2018. A convolutional fuzzy neural network for image classification. In *2018 3rd Russian-Pacific Conference on Computer Technology and Applications (RPC)* (pp. 1–4), IEEE, August.
- Le, Q. V. 2015. A tutorial on deep learning part 2: Autoencoders, convolutional neural networks and recurrent neural networks. *Google Brain* 20: 1–20.
- Lee, W. Y., S. M. Park, and K. B. Sim. 2018. Optimal hyperparameter tuning of convolutional neural networks based on the parameter-setting-free harmony search algorithm. *Optik* 172:359–67. doi:10.1016/j.ijleo.2018.07.044.
- Li, B. H., B. C. Hou, W. T. Yu, X. B. Lu, and C. W. Yang. 2017. Applications of artificial intelligence in intelligent manufacturing: A review. *Frontiers of Information Technology & Electronic Engineering* 18 (1):86–96. doi:10.1631/FITEE.1601885.
- Liang, J., and R. Liu. 2015. Stacked denoising autoencoder and dropout together to prevent overfitting in deep neural network. In *2015 8th International Congress on Image and Signal Processing (CISP)* (pp. 697–701), IEEE, October.
- Lindfield, G., and J. Penny. 2017. *Introduction to Nature-Inspired Optimization*. Waltham, MA, United States: Academic Press.
- Ma, N., X. Zhang, H. T. Zheng, and J. Sun 2018. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 116–131).
- Mallawaarachchi, V. 2017. Introduction to genetic algorithms. Online Accessed April 2, 2020. <https://towardsdatascience.com/introduction-to-genetic-algorithms-including-example-code-e396e98d8bf3>
- Masko, D., and P. Hensman. 2015. The impact of imbalanced training data for convolutional neural networks. Degree Project in Computer Science, KTH Royal Institute of Technology.
- MathWorks-1. Convolutional neural network. Online Accessed October 7, 2019. <https://uk.mathworks.com/solutions/deep-learning/convolutional-neural-network.html> .
- MathWorks-2. Deep learning using bayesian optimization. Online Accessed April 4, 2020. <https://www.mathworks.com/help/deeplearning/ug/deep-learning-using-bayesian-optimization.html> .
- MathWorks-3. Bees algorithm (BeA) in MATLAB. Online Accessed April 9, 2020. <https://uk.mathworks.com/matlabcentral/fileexchange/52967-bees-algorithm-bea-in-matlab> .
- McDermott, J. 2021. Convolutional neural networks — image classification w Keras. Online Accessed November 15, 2021. <https://www.learndatasci.com/tutorials/convolutional-neural-networks-image-classification> .

- Motepe, S., A. N. Hasan, and R. Stopforth. 2019. Improving load forecasting process for a power distribution network using hybrid AI and deep learning algorithms. *IEEE Access* 7:82584–98. doi:10.1109/ACCESS.2019.2923796.
- Ouf, H. 2017. Maxpooling vs minpooling vs average pooling. Online Accessed November 15, 2021. <https://hany-ouf.blogspot.com/2020/08/maxpooling-vs-minpooling-vs-average.html>.
- Packianather, M. S., A. K. Al-Musawi, and F. Anayi. 2019. Bee for mining (B4M)—A novel rule discovery method using the Bees algorithm with quality-weight and coverage-weight. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science* 233 (14): 5101–5112.
- Packianather, M. S., B. Yuze, E. Mastrocinque, F. Fruggiero, D. T. Pham, and A. Lambiase. 2014. Novel genetic bees algorithm applied to single machine scheduling problem. In 2014 World Automation Congress (WAC), IEEE, Waikoloa, HI, United States, April.
- Pan, H. 2017. A Study on Deep Learning: Training, Models and Applications.
- Panwar, M., J. Padmini, A. Acharyya, and D. Biswas. 2017. Modified distributed arithmetic based low complexity CNN architecture design methodology. In 2017 European Conference on Circuit Theory and Design (ECCTD), IEEE, Catania, Italy, September.
- Popko, E. A., and I. A. Weinstein. 2016. Fuzzy logic module of convolutional neural network for handwritten digits recognition. In *Journal of Physics: Conference Series*, Vol. 738, No. 1 IOP Publishing, Athens, Greece, August.
- Qolomany, B., M. Maabreh, A. Al-Fuqaha, A. Gupta, and D. Benhaddou. 2017. Parameters optimization of deep learning models using particle swarm optimization. In 2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC), IEEE, Bangkok, Thailand, June.
- Rashid, T. A., and S. M. Abdullah. 2018. A hybrid of artificial bee colony, genetic algorithm, and neural network for diabetic mellitus diagnosing. *ARO-The Scientific Journal of Koya University* 6 (1):55–64. doi:10.14500/aro.10368.
- Shah, A., E. Kadam, H. Shah, S. Shinde, and S. Shingade. 2016. Deep residual networks with exponential linear unit. In *Proceedings of the Third International Symposium on Computer Vision and the Internet*, Jaipur, India, September.
- Singh, A. K., B. Ganapathysubramanian, S. Sarkar, and A. Singh. 2018. Deep learning for plant stress phenotyping: Trends and future perspectives. *Trends in Plant Science* 23 (10):883–98. doi:10.1016/j.tplants.2018.07.004.
- Sinha, T., B. Verma, and A. Haidar. 2017. Optimization of convolutional neural network parameters for image classification. In 2017 IEEE Symposium Series on Computational Intelligence (SSCI), Honolulu, HI, United States, IEEE.
- Tch, A. 2017. The mostly complete chart of neural networks, explained. Online Accessed October 7, 2019. <https://towardsdatascience.com/the-mostly-complete-chart-of-neural-networks-explained-3fb6f2367464>.
- Wang, J., Y. Ma, L. Zhang, R. X. Gao, and D. Wu. 2018. Deep learning for smart manufacturing: Methods and applications. *Journal of Manufacturing Systems* 48:144–56. doi:10.1016/j.jmsy.2018.01.003.
- Wu, B., Z. Liu, Z. Yuan, G. Sun, and C. Wu. 2017. Reducing overfitting in deep convolutional neural networks using redundancy regularizer. In *International Conference on Artificial Neural Networks*, Springer, Cham, September.
- Wu, J. 2017. *Introduction to convolutional neural networks*. China: National Key Lab for Novel Software Technology, Nanjing University. 5, 23.
- Wuest, T., D. Weimer, C. Irgens, and K. D. Thoben. 2016. Machine learning in manufacturing: Advantages, challenges, and applications. *Production & Manufacturing Research* 4 (1):23–45. doi:10.1080/21693277.2016.1192517.

- Xu, F., C. M. Pun, H. Li, Y. Zhang, Y. Song, and H. Gao (2019). Training feed-forward artificial neural networks with a modified artificial bee colony algorithm. *Neurocomputing*
- Yamashita, R., M. Nishio, R. K. G. Do, and K. Togashi. 2018. Convolutional neural networks: An overview and application in radiology. *Insights into Imaging* 9 (4):611–29. doi:[10.1007/s13244-018-0639-9](https://doi.org/10.1007/s13244-018-0639-9).
- Zeybek, S., D. T. Pham, E. Koç, and A. Seçer. 2021. An improved bees algorithm for training deep recurrent networks for sentiment classification. *Symmetry* 13 (8):1347. doi:[10.3390/sym13081347](https://doi.org/10.3390/sym13081347).
- Zhang, H., S. Kiranyaz, and M. Gabbouj. 2018. Finding better topologies for deep convolutional neural networks by evolution. arXiv preprint arXiv:1809.03242.