International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 03 | Mar 2020 www.irjet.net p-ISSN: 2395-0072

Implication of Convolutional Neural Network in the Classification of Vitiligo

Varinder Kantoria[1], Shubham Sharma[2], Shashank Bhushan[3], Harshit Saini[4], Rahul Nijhawan[5]

[1] Student, Dept. of CSE, Roorkee Institute of Technology, Roorkee, Uttarakhand, India
[2]Student, Dept. of CSE, Quantum School of Technology, Roorkee, Uttarakhand, India
[3]Student, Dept. of CSE, College of Engineering Roorkee, Uttarakhand, India
[4]Student, Dept. of CSE, College of Engineering Roorkee, Uttarakhand, India
[5] Assistant Professor, Dept. of CSE, Graphic Era Deemed-to-be University, Dehradun, Uttarakhand, India

---***---

Abstract - *In recent years, machine learning has been successfully applied in the field of medical sciences to diagnose the various diseases by image classification and has achieved enormous success for diagnosing skin diseases like psoriasis, chronic dermatitis, vitiligo, hives, rosacea, etc. In particular, vitiligo is a complex pigmentary disorder of the skin, characterized by the destruction of melanocytes, apparently marked by white patches on the skin. In the present study, classification based on the deep learning-based approach to solving the problem of classifying Vitiligo lesions using convolutional neural networks (CNNs) was employed. We have used the dataset of 696 images having 368 vitiligo infected images and 328 normal images to carry out the whole problem. At the technical facet, we have used four pre-trained models for feature extraction namely, Inception-V3, VGG-16, VGG-19, and SqueezeNet. Using these pre-trained models, we have trained four classification models, kNN (k-nearest neighbors), SVM (Support Vector Machine), Convolutional Neural Network and Logistic Regression. Our findings revealed that Inception-V3 provides maximum accuracy of 98.0% on Logistic Regression and 96.5% on Neural Network.*

***Key Words*:** **Vitiligo, Convolutional Neural Network (CNN), SVM, KNN, Logistic Regression, Classification**.

1.INTRODUCTION

Vitiligo is an acquired, idiopathic multifactorial pigmentary disorder of the skin, clinically characterized by white marks or patches due to selective melanocyte damage [1]. The pathophysiology of vitiligo is unclear yet, however, variable mechanisms including oxidative stress, metabolic abnormalities, inflammatory stress, autoimmune, neural or genetic conditions may contribute to the development of vitiligo [2]. The worldwide prevalence of disease ranges from 0.38-0.3.2 %, mostly affecting at childhood, wherein, the prevalence varies from ⅓ to ½ of all the cases [3]. In an Indian cohort study, it has been reported that approximately 56.7 % of cases belong to an age group of 8-12 years and a majority of cases have been reported from Gujarat, accounting for 8.8% of worldwide prevalence [4].

Now-a-days machine-based diagnosis system have been employed for the classification of this disease. In this field, few reports have been present in the literature and still more need to be explored in this regard. Deep convolutional neural network model like convolutional neural networks (CNN) have been widely used for the classification of vitiligo nowadays [5,6]. However, CNN is little different from the regular neural networks. In CNN, the layers are organized in 3 dimensions viz. height, depth, and width and in one layer, the neurons do not make connections with the neurons of next layer but a small region only and the last output reduced to a single vector of probability scores, well ordered along the depth dimension[7,8]. The collection of datasets is always a hurdle in such type of operations, but we used Kaggle repository for the dataset collection, wherein, there were 368 images of vitiligo infected body parts. Thereafter, we organized the dataset for the classification and for the control we used 328 healthy body images for differentiation from the google. For the analysis, we have divided our dataset into two classes, healthy and vitiligo infected. In this, 70% images were used for the training part and remaining 30% images were used for the testing part. We have trained the CNN multi-class model on the 70% images for disease classification and lesion targeted classification. Further, four pre-trained models were used for feature extraction, Inception-V3 [9], VGG-16 [10], VGG-19[11], and SqueezeNet [12]. The feature extraction models transform the images into a reduced set of variables comprised of important and crucial information only. After feature extraction by these models, we have used classification Algorithms KNN, SVM [13], CNN and Logistic Regression. All the classification algorithms were used with all the pre-trained models in a sequential manner. Firstly, we applied inception-V3 for feature extraction and then all four algorithms were applied. The results confirmed that our diagnosis system showed 98.0% accuracy on Logistic regression, using inception-V3 model, 97.7% accuracy on Neural Network and KNN (K-nearest neighbor) using VGG-16.

2. DATASET DESCRIPTION

Data collection is the most important aspect of the diagnosis system and selection of appropriate sample is very crucial for such experiments in machine learning. For the experiment, we have collected vitiligo infected lesions from one of the largest data science repositories, Kaggle database, that is already in the public domain and normal body images from google sources in order to train the models for better results.

On the basis of which we have categorized them into two classes, one is vitiligo infected lesions and another is the normal body parts. In the dataset, there were 368 images of vitiligo infected lesions and 328 images of normal body parts are shown in figure 1 as given below.

Fig-1: Sample images of Vitiligo infected body images and normal body images

Furthermore, the dataset was divided into two parts in which 70% data was used for training and 30% was used for testing purpose.

3.METHOD

In order to classify the vitiligo infected and normal body images, we have used the promising architecture known as convolutional neural network, recently known as a popular algorithm in feature learning and object classification. The machine used for performing this classification is configured for 8GB of RAM, 4GB of GPU (Nvidia) with 8 cores, and an Intel i7 processor.

In this experiment, we have used four pre-trained models for feature extraction, Inception-V3, VGG-16, VGG-19, and SqueezeNet. The main feature of these feature extraction models is to transform the images into a reduced set of variables comprised of important and crucial information only, for conducting experiments accordingly.

First, we employed the Inception-V3 model which is a powerful model developed at Google and pre-trained on ImageNet, a large dataset of web images (1.4M images and 1000 classes). Along with feature extraction, it also does classification with SoftMax and fully-connected layers [9]. Using these extracted features, we have used data mining functions to achieve the goal of classification to accurately predict the target classes and for that we have trained four classification models, kNN (k-nearest neighbors) [14, 15], SVM(support vector machine), convolutional neural network and logistic regression and we found maximum accuracy of 98.0% with logistic regression and 96.5% with neural network. Second, we have used SqueezeNet, which is a small DNN architecture that achieves AlexNet-level accuracy on ImageNet within 50x parameters. we compressed SqueezeNet below 0.5 MB (510x smaller than AlexNet). SqueezeNet works as similar as a fully-connected layer that works on feature points in the same position and lowers the depth of a feature map [12]. After performing all four classification models like inception-V3, SqueezeNet gave an accuracy of 97.4% on CNN (Convolutional Neural Network) and SVM (Support Vector Machine).

Third, we have used the VGG-16 model which refers to the VGG model (proposed by the Visual Geometry Group in the University of Oxford) with 16 weight layers. The input layer takes an image of the size (224 x 224 x 3), and the output layer is a SoftMax prediction on thousands of classes [10]. The feature extraction part of model extends from the input layer to last max-pooling layer (labeled by 7 x 7 x 512) and rests as a classification part of the model. All four models already used above were employed for VGG-16 and results revealed that we got an accuracy of 97.7% on kNN (k-nearest neighbors) and CNN (Convolutional Neural Network).

At last, we have used the VGG-19 model, which is a CNN model made of 19 layers having been trained on millions of image samples and utilizes the architectural style of zero-center normalization on images convolution, ReLU, Max Pooling, etc. [11]. All the above classifiers were used for classification, the VGG-19 model has achieved maximum accuracy of 98.0% on Logistic Regression and 97.4% on Neural Network.

3.1 CNN (Convolutional Neural Network)

Convolutional Neural Network (CNN) operates from a mathematical perspective and is a regularized variant of a class of feedforward artificial network (ANN) known as multilayer perceptron's that generally means fully connected networks in which every neuron in a layer is connected to all neurons in the further layers [7]. Regularization applies to objective functions in ill-posed optimization problems and adds on the information in order to solve an ill-posed problem or to prevent overfitting [5].

Now, we'll see how CNN trains and predicts in the abstract level so, When it comes to programming a CNN, it usually takes an order 3 tensor as input with shape (no. of mages) x (image width) x(image depth) that sequentially goes through a series of processing like convolutional layer, a pooling layer, a normalization layer, a fully connected layer, a loss layer, etc. as shown in figure 2. that makes abstracted images to a feature map, with the shape (no. of images) x (feature map width) x (feature map Channels) [7,16].

Here, tensors are just higher-order matrices and below we have given layer by layer running of CNN in a forward pass:

$$x^1 \rightarrow w^1 \rightarrow x^2 \rightarrow \ldots . \rightarrow x^{L-1} \rightarrow w^{L-1} \rightarrow x^L \rightarrow w^L \rightarrow z^L$$

Where x^1 usually an image (order 3 tensor). It goes through the processing in the first layer, w^1 denotes parameters involved in the first layer's processing collectively as a tensor w^1. The output of the first layer is x^2, which also acts as the input to the second layer processing and the same follows till all layers in the CNN have been finished, which outputs x^L. To make x^L a probability mass function, we can set the processing in the (L-1) th layer as a SoftMax transformation of x^{L-1} (cf. the distance metric and data transformation notes) last layer is the loss layer. Let us suppose here t that is the corresponding target (ground-truth) value for the input x^1, then a cost or loss function can be used to measure the discrepancy between the CNN prediction x^L and the target t, for which a simple loss function can be given as follows:

$$z = \frac{1}{2}\left|\left|t - x^L\right|\right|^2$$

Now coming to the ReLU Layer it does not change the size of the input, that is, xl and y share the same size. In fact, the Rectified Linear Unit (ReLU) can be regarded as a truncation performed individually for every element in the input:

$$y_{a,b,d} = \max\{0, x^l_{a,b,d}\}$$

with $0 \le a < H^l = H^{l+1}$, $0 \le b < W^l = W^{l+1}$, and $0 \le d < D^l = D^{l+1}$.

There is no parameter inside a ReLU layer, hence we have no need for parameter learning in this layer. Based on the above equation it can be given as:

$$\frac{dy_{a,b,d}}{dx^l_{a,b,d}} = [[dx^l_{a,b,d} > 0]]$$

Being 1 if its argument is true, and 0 otherwise. Hence, we have

$$\left[\frac{\partial z}{\partial x^l}\right] = \left\{[\frac{\partial z}{\partial y}]_{a,b,d} \ \ if \ x^l_{a,b,d} > 0\right\}$$

$$\left[\frac{\partial z}{\partial x^l}\right] = \{0 \qquad otherwise\}$$

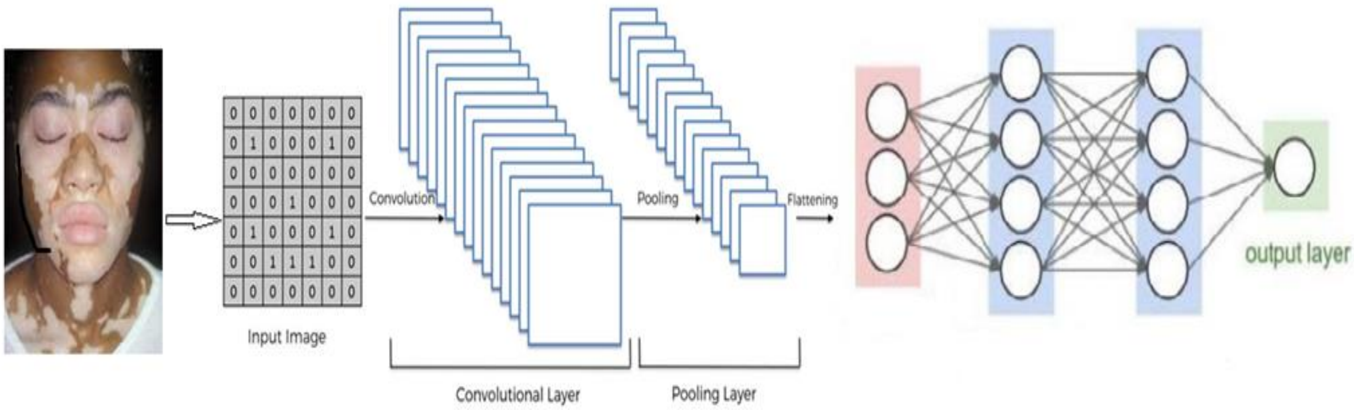

Fig-2: Working of Convolutional Neural Network

3.2 KNN (K-nearest neighbors)

In the KNN (K-nearest neighbors) classification method, the class label of a test sample is determined based on the k nearest samples from the training dataset. In this, we need to separate the training dataset into different clusters, for which K-means clustering algorithm was used [14]. For detecting the k nearest neighbors, the distance should be calculated between the test samples and all training ones [15]. For finding the distance, Euclidean method is used

$$d_i^a = max\left(\sqrt{p_b^2 - (C_b^a)^2}\right) \forall P \epsilon C_a$$

Here, d_b^a is the distance (Euclidean distance) between the data points of a^{th}cluster and its center C_a along the b^{th} axis.

The idea behind this is to describe a metric that at least approximation shows the distance of a test data sample to a cluster border [15]. For more accurate approximation, the distance is computed by both the cluster's spread on different axes and the data point coordinates as follows:

$$D^a = d^a - \sum_{i=1}^{n} \frac{\sqrt{x_b^2 - (C_b^a)^2}}{d^l} \times d_b^a$$

Here, n is the number of dimensions in the space of data, x is the test sample, d^a is the Euclidean distance between the data points of a^{th}cluster and it's center C_a along the b^{th} axis and D^a is the approximate distance of the test sample from a^{th}cluster border [15].The output depends on the training dataset used. It can have a considerable effect on the final classification process to select the best cluster of the data. As a result, this algorithm is applied to the selected part of training data to find the k nearest neighbors and therefore, to estimate the test sample's class.

3.3 Logistic Regression

Logistic regression is a classification algorithm in a machine learning. Logistic regression is based on the concept of probability. Logistic regression logistic regression transforms its output and returns a probability value by using a sigmoid function [17]. Let us consider a Logistic model with two predictors a_1, a_2 and one Bernoulli response variable Z denoted as p = P(Z=1) [18]. This linear relationship between the log-odds of the event i.e. Z=1 and the predictor variables can be written in mathematical form as follows:

$$l = log_b \frac{p}{1-p} = \beta_0 + \beta_1 x_1 + \beta_2 x_2$$

Where l is the log-odds,β_i are parameters of the model and b is the base of the logarithm [18]. The odds can be recovered by exponentiating the log-odds:

$$\frac{p}{1-p} = b^{\beta_0 + \beta_1 x_1 + \beta_2 x_2}$$

By algebraic manipulation, the probability Z=1 is:

$$p = \frac{b^{\beta_0 + \beta_1 x_1 + \beta_2 x_2}}{b^{\beta_0 + \beta_1 x_1 + \beta_2 x_2} + 1} = \frac{1}{1 + b^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2)}}$$

By the above formula when $\beta_i$are fixed, either the probability that Z=1 for a given observation or the log-odds that Z=1 for a given observation can be easily computed. In a logistic model [17,18], the main use-case is to be given the probability p that Z=1 and an observation (x_1, x_2).

3.4 SVM (Support Vector Machine)

Support Vector Machine (SVM) is a linear model for regression and classification of the problems. It efficiently works for many linear and nonlinear practical problems [13]. The SVM algorithm creates a hyperplane or a line which separates the data into classes [19]. Support vectors are the data points that lie closest to the hyperplane. SVMs maximize the margin around the separating hyperplane.

The decision function is fully specified by a subset of training samples, the support vectors. Support Vectors are the elements of the training set and are critical elements of the training set that would change the position of the dividing hyperplane if removed [19]. The problem of finding the optimal hyperplane is an optimization problem that can be solved by optimization techniques for which we have used Lagrange multipliers to get this problem into a form that can be solved analytically as given below:

Lagrangian Formulation:

$$\min L_p = \frac{1}{2}\left||w|\right|^2 - \sum_{i=1}^{l} a_i y_i (x_i . w + b) + \sum_{i=1}^{l} a_i$$

such that $\forall i, a_i \geq 0$ where l is the no. of training points

From the property that the derivatives at min=0, we get:

$$\frac{\partial L_p}{\partial w} = w - \sum_{i=1}^{l} a_i y_i x_i = 0$$

$$\frac{\partial L_p}{\partial b} = \sum_{i=1}^{l} a_i y_i = 0$$

so, $w = \sum_{i=1}^{l} a_i y_i x_i, \sum_{i=1}^{l} a_i y_i = 0$

We have calculated the ROC (Receiver Operating Characteristics) curve as given in figure 3 for mapping the performance of our experiment, where on X-axis specificity is shown and on Y-axis Sensitivity is defined. ROC curve on Vitiligo Classification is shown in figure 3:

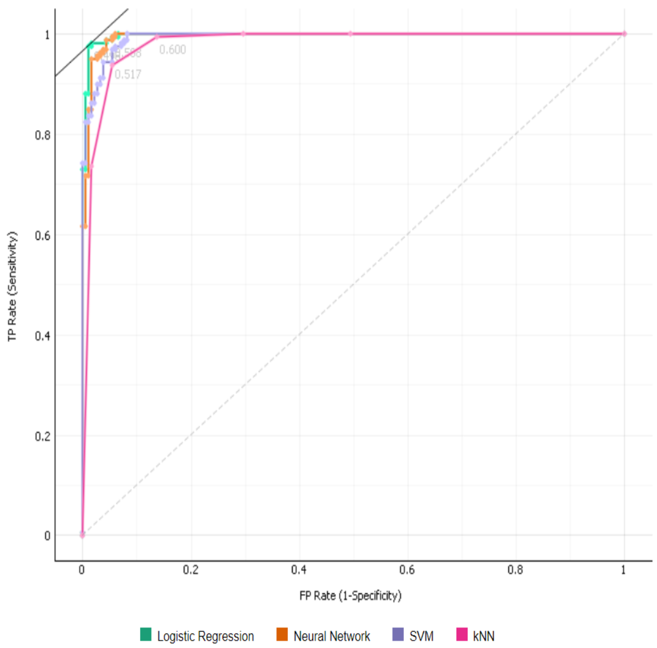

Fig-3: ROC curve of vitiligo classification

4. RESULT AND DISCUSSION

In this context, we have discussed our results that we have got after performing the whole experiment over the classification of vitiligo lesions by using four pre-trained models for feature extraction named as Inception-V3, VGG-16, VGG-19, and SqueezeNet and using these pre-trained models we have trained four classification models for contrast that are kNN (k-nearest neighbors), SVM(Support Vector Machine), Convolutional Neural Network and Logistic Regression.

First, In Inception-V3 we got a maximum accuracy of 98.0% on Logistic Regression and 96.5% on Neural Network. Second, In SqueezeNet we got an accuracy of 97.4% on CNN (Convolutional Neural Network) and SVM (Support Vector Machine). Third, In VGG-16 we got an accuracy of 97.7% on kNN (k-nearest neighbors) and CNN (Convolutional Neural Network). At last, the VGG-19 model has achieved maximum accuracy of 98.0% on Logistic Regression and 97.4% on Neural Network. Respective accuracy assessment of vitiligo classification on each model is shown below in accuracy table:

Table-1: Accuracy Estimation

Feature Extraction	Model	AUC	Accuracy	Precision	Recall
Inception-V3	KNN	0.980	0.924	0.933	0.924
	SVM	0.993	0.947	0.947	0.947
	Neural Network	0.994	0.965	0.966	0.965
	Logistic Regression	0.997	0.980	0.980	0.980
SqueezeNet	KNN	0.986	0.939	0.940	0.939
	SVM	0.997	0.974	0.974	0.974
	Neural network	0.993	0.974	0.974	0.974
	Logistic Regression	0.995	0.968	0.968	0.968
VGG-16	KNN	0.994	0.977	0.977	0.977
	SVM	0.996	0.924	0.929	0.924
	Neural Network	0.998	0.977	0.977	0.977
	Logistic Regression	0.998	0.971	0.971	0.971
VGG-19	KNN	0.991	0.944	0.945	0.944
	SVM	0.994	0.930	0.932	0.930
	Neural Network	0.998	0.974	0.974	0.974
	Logistic Regression	0.998	0.980	0.980	0.980

5. CONCLUSION

In the present study, all the used approaches were executed on available datasets to diagnose the vitiligo infected lesions for assisting dermatologists to diagnose the disease at early stage. We have implemented a classification mechanism that builds a model using deep CNN which predicts whether a lesion is vitiligo infected or normal. The proposed model has given a promising and imperative performance that accounts in the field of deep learning. The method is more effective and robust with great reliability. Based on pre-processing and feature extraction, feature selection, classifier combination in which the same input (features) is applied to different natures of classifiers using different pre-trained models with the development of deep-learning architecture has given enormous accuracy of 97.7% on Neural Network and 98.0% on Logistic Regression model. In this mechanism, the diagnostic ability of a binary classifier system is obtained in terms of receiver operating characteristics (ROC) and feature selection approaches also built to employ fewer features (input) to represent data and to lower the computational cost, without contorting discriminative capabilities.

It also improved learner accuracy, domain understandability and, decreased model complexity etc. Based on our obtained results many other aspects can also be considered as the future directions for continuing the research on vitiligo disease. The developed mechanism can be further expanded for the future perspective in neural networks to help in reducing difficulties for the doctors.

6. ACKNOWLEDGEMENT

I wish to express my deepest gratitude to Dr. Ankush Mittal who assisted us in the field of Machine Learning and helped to bring all the research together into a unified theme with their competent advice and experience. It is whole-heartedly appreciating that their great advice for our work proved extremely monumental towards the success of this study. Another name to acknowledge here is Dr. Shiv Kumar who has presented their valuable contribution to the support and guidance for writing a research paper that is worth appreciating.

7. REFERENCES

1. Arora, Amanjot Kumaran, Muthu - Pathogenesis of vitiligo: An update Y1 - 2017/7/1 JF - Pigment International JO - Pigment Int SP - 65 EP - 77 VL - 4 IS - 2 UR.
2. Zhang, Yuhui AU - Cai, Yunfei AU - Shi, Meihui AU - Jiang, Shibin AU - Cui, Shaoshan AU - Wu, Yan AU - Gao, Xing-Hua AU - Chen, Hong-Duo T1 - The Prevalence of Vitiligo: A MetaAnalysis LA - eng SN - 1932-6203 Y1 - 2016/09/27
3. Daniel, S. J., & Sivanesan, A. R. (2017). Dermatological quality of life and psychiatric morbidity among 200 vitiligo patients. Int J Sci Study, 5(21), 86-92.
4. Ezzedine, K., & Silverberg, N. (2016). A practical approach to the diagnosis and treatment of vitiligo in children. Pediatrics, 138(1), e20154126.
5. Haofu Liao, Yuncheng Li, and Jiebo Luo, "Skin disease classification versus skin lesion characterization: Achieving robust diagnosis using multi-label deep neural networks," 2016 23rd International Conference on Pattern Recognition (ICPR), Cancun, 2016, pp. 355-360.A Practical Approach to the Diagnosis and Treatment of Vitiligo in Children JF - Pediatrics JO - Pediatrics.
6. N. C. F. Codella et al., "Deep learning ensembles for melanoma recognition in dermoscopy images," in IBM Journal of Research and Development, vol. 61, no. 4/5, pp. 5:1-5:15, 1 July-Sept. 2017.
7. Liu J., Yan J., Chen J., Sun G., Luo W. (2019) Classification of Vitiligo Based on Convolutional Neural Network. In: Sun X., Pan Z., Bertino E. (eds) Artificial Intelligence and Security. ICAIS 2019. Lecture Notes in Computer Science, vol 11633. Springer, Cham.
8. R. Nijhawan, R. Verma, Ayushi, S. Bhushan, R. Dua and A. Mittal, "An Integrated Deep Learning Framework Approach for Nail Disease Identification," 2017 13th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), Jaipur, 2017, pp. 197-202.
9. Xiaoling Xia, Cui Xu, and Bing Nan, "Inception-v3 for flower classification," 2017 2nd International Conference on Image, Vision, and Computing (ICIVC), Chengdu, 2017, pp. 783-787.
10. Alfredo Canziani and Adam Paszke and Eugenio Culurciello, An Analysis of Deep Neural Network Models for Practical Applications,2016.
11. Koesdwiady A., Bedawi S.M., Ou C., Karray F. (2017) End-to-End Deep Learning for Driver Distraction Recognition. In: Karray F., Campilho A., Cheriet F. (eds) Image Analysis and Recognition. ICIAR 2017. Lecture Notes in Computer Science, vol 10317. Springer, Cham.
12. Forrest N. Iandola and Song Han and Matthew W. Moskewicz and Khalid Ashraf and William J. Dally and Kurt Keutzer, SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size,2016.
13. Y. Bazi and F. Melgani, "Toward an Optimal SVM Classification System for Hyperspectral Remote Sensing Images," in IEEE Transactions on Geoscience and Remote Sensing, vol. 44, no. 11, pp. 3374-3385, Nov. 2006.
14. Zhang, Shichao and Li, Xuelong and Zong, Ming and Zhu, Xiaofeng and Cheng, Debo, Learning k for KNN Classification,2017, Association for Computing Machinery, New York, NY, USA.
15. Zhenyun Deng, Xiaoshu Zhu, Debo Cheng, Ming Zong, Shichao Zhang, Efficient kNN classification algorithm for big data, Neurocomputing, Volume 195,2016.
16. M. Mustafa, M. N. Taib, Z. H. Murat, N. Sulaiman, and S. A. M. Aris, "The Analysis of EEG Spectrogram

Image for Brainwave Balancing Application Using ANN," 2011 UkSim 13th International Conference on Computer Modelling and Simulation, Cambridge, 2011, pp. 64-68.

17. Liu, Yuan Y. Yang, Min Ramsay, Malcolm Li, Xiao S. Coid, Jeremy W.2011/12/01, A Comparison of Logistic Regression, Classification and Regression Tree, and Neural Networks Models in Predicting Violent Re-Offending Journal of Quantitative Criminology.
18. Imran Kurt, Mevlut Ture, A. Turhan Kurum, Comparing performances of logistic regression, classification and regression tree, and neural networks for predicting coronary artery disease, Expert Systems with Applications,2008.
19. A. Mathur and G. M. Foody, "Multiclass and Binary SVM Classification: Implications for Training and Classification Users," in IEEE Geoscience and Remote Sensing Letters, vol. 5, no. 2, pp. 241-245, April 2008.
20. R. Nijhawan, H. Sharma, H. Sahni and A. Batra, "A Deep Learning Hybrid CNN Framework Approach for Vegetation Cover Mapping Using Deep Features," 2017 13th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), Jaipur, 2017, pp. 192-196.
21. Nijhawan R., Joshi D., Narang N., Mittal A., Mittal A. (2019) A Futuristic Deep Learning Framework Approach for Land Use-Land Cover Classification Using Remote Sensing Imagery. In: Mandal J., Bhattacharyya D., Auluck N. (eds) Advanced Computing and Communication Technologies. Advances in Intelligent Systems and Computing, vol 702. Springer, Singapore.
22. R. Nijhawan, I. Srivastava and P. Shukla, "Land cover classification using super-vised and unsupervised learning techniques," 2017 International Conference on Computational Intelligence in Data Science (ICCIDS), Chennai, 2017, pp. 1-6.
23. D. Chandra, S. S. Rawat and R. Nijhawan, "A Machine Learning Based Approach for Progeria Syndrome Detection," 2019 4th International Conference on Information Systems and Computer Networks (ISCON), Mathura, India, 2019, pp. 74-78.