iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 8 **Issue:** III **Month of publication:** March 2020

DOI:

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue III Mar 2020- Available at www.ijraset.com

A Leukonychia Nail Disease Detection using Deep Learning Approach

Naveen[1], Gaurav Rai[2], Aquib Hussain[3], Amit Kumar[4], Rahul Nijhawan[5]
[1, 2, 3, 4]Student, COER Roorkee, Uttarakhand, India
[5]Guide, GEU Dehradun, Uttarakhand, India

Abstract: Nail Diseases are some kind of deformities in nails. Detection of Nail Disease (Leukonychia) using modern state-of-the-art technological tools, thereby reducing the dependency on medical experts. It is still challenging to identify the disease in Nails. Leukonychia, is a medical term for white discoloration appearing on nails. The most common cause is injury to the base of the nail where the nail is formed. The common causes that are noticed around the nails are changing the color of nails to dark streaks, white streaks, changes in nail shape, changes in nail thickness, bleeding around nails or change in nail color. This affection of the disease is about 31% in the adult population. Treatment of leukonychia has been reported from treating malnutrition associated with zinc deficiency and other minerals. This paper strives for a deep convolutional neural network to classify Leukonychia disease from images that are provided in the dataset. The skeleton uses (VGG)-16, (VGG)-19 and INCEPTION-V3 for extracting the features. Due to the absence of an active dataset, a new dataset was created for testing the correctness of our debatable framework. This work is tested on our dataset and is measured with other advanced algorithms (Artificial Neural Network, Support Vector Machine, Random Forest, Adaboost, CN2 Rule Induction, K-Nearest Neighbour, Naive Bayes and Logistic Regression) which results in considerable performance in extraction of features.

Keyword: Nail Disease, Deep Convolutional Neural Network, Feature Extraction etc.

I. INTRODUCTION

Keratin is the building blocks of nails, which strengthens the tissues of nails and fingers. By looking at the nails we can detect patients' health conditions. Some variations in nail colors, malignity, nail appearances, and presence of some particular problems tell us about diseases like diabetes, fungal infections, liver diseases, melanoma, and malnutrition [10]. Anybody can possess these nail disorders. Latest research shows that around 25% of people possess leukonychia (a rare disorder), which happens because of vitamin deficiency. It can be detected by observing the common changes in nails such as changing the color of nails to white, thickening of nails, changes in nail shape, swelling or redness around nails [2].

Nail Disease Detection Framework is gaining importance of late. A lot of researchers are interested in the field. Dermatology is a branch of the medical field which is attracting a lot of researchers [2]. Image comparing methodology, automatic melanoma detection, classification of dermatological images, and skin disease prediction are the latest trends. Deep Learning algorithms provided results as set side by side to the traditional methods like Logistic Regression (LR), Support Vector Machine (SVM), Random Forest (RF), K-Nearest Neighbour (KNN), Artificial Neural Network (ANN), Naive Bayes(NB), Adaboost and CN2 Rule Induction. [8] The problems detected by our algorithm is the scarcity of high resolution images set for this process. We required the exact and standard set of images for computing the result. Regarding these liabilities we have combated modern machine learning techniques and also uses deep Convolutional Neural Networks. The training data created by us subsist of 200–280 images in which around 65% data have been chosen haphazardly for training data and remaining 35% for the purpose of testing. Assessing success rate is achieved via classification success, Area Under Curve (AUC) is calculated together with Receiver Operating Characteristic (ROC) curves. This research paper is arranged according to these criteria: (Section 1) Introduction, (Section 2) Literature Review, (Section 3) Data Collection, (Section 4) Methodology, (Section 5) Proposed Approach, (Section 6) Results and Discussions, and (Section 7) Conclusion.

II. LITERATURE REVIEW

Currently, many scientists are working in disease identification techniques like extraction of features and allocation. To create an exact arrangement of Leukonychia disease from an image dataset, a lot of researchers have pretended to apply both deep learning and machine learning algorithms recently [6]. This section briefly narrates the various studies associated with arranging Leukonychia from image dataset by applying deep learning approaches. Categorization of leukonychia may be accomplished precisely by implementing various approaches like ML and DL models, which have been continued in recent years [7]. These techniques contain ANN, CNN, RF, SVM, NB, LR, CN2 Rule Induction and Adaboost [8]. This section narrates the process of executing various Imitations for Categorization of leukonychia from image dataset.

Nijhawan et al. [2] used AlexNet structuring, that is trained for Categorization of different disease of nails. Being precise, RELU is applied as a sigmoid function to get excellent accuracy. Pandit et al. [4] studied features of image and prophecies disease using information of Medical palmistry. Thus, our analysis comes up with a DL skeleton for Leukonychia identification in the beginning stage which is obtaining the accuracy of 93.8%.

III. DATA COLLECTION

The collection of data is a significant part which involves some of the toughest machine learning questions. Input to our algorithm involves clear images as our training data for enabling the training process of our models. To increase the success rate of our result after testing we need to train more by using various models [7].

The problem faced by us was no proper open source for our dataset. Because of this concern we had decided to make our own dataset of images of 200-280 images, of varying diseases. After this the data is transferred to our architecture and then our pre-trained model uses fine tuning on Alexnet.

We have selected high resolution images showing unaffected and the affected nails. We used CNN for extraction of features (Fig. 1).

Fig. 1. Illustration of disease

We divided our training dataset for further training and testing. In this dataset, we randomly selected 65% of the images in each category to make a training dataset and the rest 35% were used to create a test dataset (Table 1).

Table 1. Specification of images

Nail categories	Total number of images
True nails	108
Leukonychia	102

IV. METHODOLOGY

We created our project on a system of 8GB RAM and a GPU of NVIDIA(GEOFORCE) having 8th gen intel core i5 with 8 cores and 4 threads. Our first work was to use the dataset on pre-trained Alex Network for fine tuning the architecture [6]. After our first step we tried to compare our methodology by executing our data on a lot of algorithms with modern and powerful algorithms like Support vector machine, K-Nearest Neighbour, Logistic Regression, AdaBoost, Artificial Neural Network, CN2 Rule Induction, Naïve Bayes and Random Forest.

Some models and their parameters are given below:

A. *Logistic Regression*

Model parameters :
Regularization: Ridge (L2), C=1

B. SVM

Model parameters :
SVM type: SVM
C=1.0
ε=0.1
Kernel: RBF, exp(-auto|x-y|²)
Numerical tolerance: 0.001
Iteration limit: 100
The approach behind Naive Bayes and Logistic Regression :

C. Naive Bayes

A data point $\underline{z} = \{z_1, \ldots\ldots, z_n\}$of n features, naive Bayes predicts the class D_k for $\underline{z}$according to probability

$$p(D_k/\underline{z}) = p(D_k/z_1,\ldots\ldots,z_n) \text{for k=1,...,k}$$

Using bayes's theorem, following can be factorized as :

$$p(D_k/\underline{z})= \frac{p(\underline{z}/D_k)p(D_k)}{p(\underline{z})} = \frac{p(z_1,\ldots\ldots,z_n/D_k)p(D_k)}{p(z_1,\ldots,z_n)}$$

Applying the chain rule, the factor $p(z_1, \ldots\ldots, z_n/D_k)$in the numerator can be additionally break down as :

$$p(z_1, \ldots, z_n/D_k)= p(z_1/z_2, \ldots, z_n, D_k)p(z_2/z_3, \ldots, z_n, D_k)\ldots p(z_{n-1}/z_n, D_k)p(z_n/D_k)$$

Naive Bayes models assume that feature z_i is independent of feature $z_j$for i$\pm j$ given the class D_k. Applying the previous decomposition, this can be formulated as :

$$p(z_i/z_{i+1}, \ldots, z_n/D_k)= p(z_i/D_k) \Rightarrow p(z_1, \ldots\ldots\ldots, z_n/D_k)=\prod_{i=1}^{n} p(z_i/D_k)$$

Thus,

$$p(D_k/z_1, \ldots, z_n)\alpha\ p(D_k,z_1,\ldots\ldots,z_n)$$

$$\alpha\ p(D_k)\, p(z_1, \ldots\ldots, z_n/D_k)$$

$$\alpha\ p(D_k)p(z_1/D_k)p(z_2/D_k)\ldots p(z_n/D_k)$$

$$\alpha\ p(D_k)\prod_{i=1}^{n} p(z_i/D_k)$$

The class conditional feature $p(z_i/D_k)$ probabilities are usually modeled using the same class of probability distribution, such as the Binomial distribution or Gaussian distribution.

Naive Bayes gives the probability of a data point $\underline{z}$ belonging to class D_k as proportional to a simple product of n+1 factors (the class prior p(D_k) plus n conditional feature probabilities$p(z_i/D_k)$). Since categorization involves assigning the class $D_k$to the data point for which the value $p(D_k/\underline{z})$ is greatest, this proportional product can be used to determine the most likely class assignment. Specifically,

$$p(D_a)\prod_{i=1}^{n} p(z_i/D_a) > p(D_b)\prod_{i=1}^{n} p(z_i/D_b) \Rightarrow p(D_a/z_1,\ldots\ldots,z_n) > p(D_b/z_1,\ldots\ldots,z_n)$$

Thus, the most likely class assignment for a data point $\underline{z} = \{z_1, \ldots\ldots, z_n\}$ can be found by calculating p(D_k)$\prod_{i=1}^{n} p(z_i/D_k)$ for k=1,2,.......,k and assigning $\underline{z}$ the class D_k for which this value is largest. In mathematical notation, this is defined

$$\widehat{D}=\text{argmax } p(D_k)\prod_{i=1}^{n} p(z_i/D_k)$$

where $\widehat{D}$is the estimated class for $\underline{z}$ given its features $z_1, \ldots\ldots, z_n$.

D. Logistic Regression

Logistic Regression is a classification algorithm that works by trying to learn a function that approximates P($\frac{X}{Y}$). It makes the central assumption that P($\frac{X}{Y}$) can be approximated as a sigmoid function applied to a linear combination of input features . Mathematically, for a single training datapoint (x,y) Logistic Regression assumes.

$$p(y=1/X=x)=\sigma(z) \text{where } z=\theta_0+\sum_{i=1}^{m} \theta_i x_i$$

This assumption is often written in the equivalent forms:

$$p(y=1/X=x) =\sigma(\theta^T x)$$

$$p(y=0/X=x) = 1-\sigma(\theta^T x)$$

where we always set $x_0$to be 1 by total law of probability

We can write an equation for the likelihood of all the data . If you take the log of the likelihood equation the result is:

$$LL(\theta) = \sum_{i=1}^{n} y^{(i)} log\ \sigma(\theta^T x^{(i)})+(1-y^{(i)})\log[1-\theta^T x^{(i)})]$$

E. Gradient of Log

Here is the partial derivative of log likelihood with respect to each parameter θ_j:

$$\frac{\partial LL(\theta)}{\partial \theta_j} = \sum_{i=1}^{n} \left[y^{(i)} - \sigma(\theta^T x^{(i)})\right] x_j^{(i)}$$

The small step that we continually take given the training dataset can be calculated as:

$$\theta_j^{new} = \theta_j^{old}+\eta.\frac{\partial LL(\theta^{old})}{\partial \theta_j^{old}}$$

$$\theta_j^{old} + \eta. \sum_{i=0}^{n} \left[y^{(i)} - \sigma(\theta^T x^{(i)})\right] x_j^{(i)}$$

η= magnitude of the step size.

V. PROPOSED APPROACH

We applied Deep Convolutional Neural Network . In the Convolutional Neural Network, images are provided as input to the image classifier , which processes the image and classifies the images under numerous classes [4]. In the Convolutional Neural Network, it has a lot of Hidden layers, our first operation on the convolution of an array of pixels of images [9]. We passed our output generated in the previous step to the next layer that is responsible for amplitude reduction of the data set of images for overcoming the difficulty of misfit. In the second step we analyse and categorize (Leukonychia) nail disease.(Fig. 2)

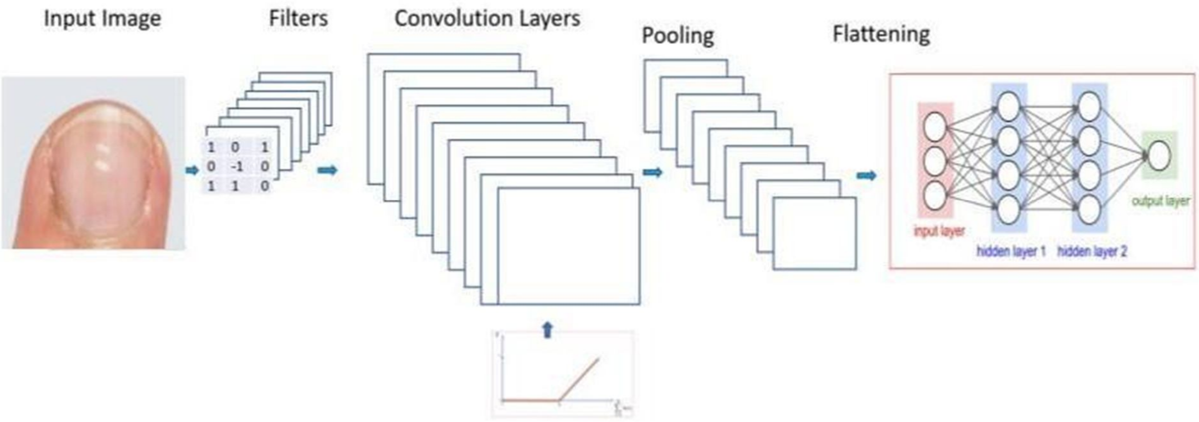

Fig. 2. Working of Convolutional Neural Network

Our approach involves deep learning which uses 19-layer CNN strategy trained over thousands of images on our database [7]. The previous output is given to ANN that classifies the leukonychia affected as well as unaffected nails (Fig. 3).

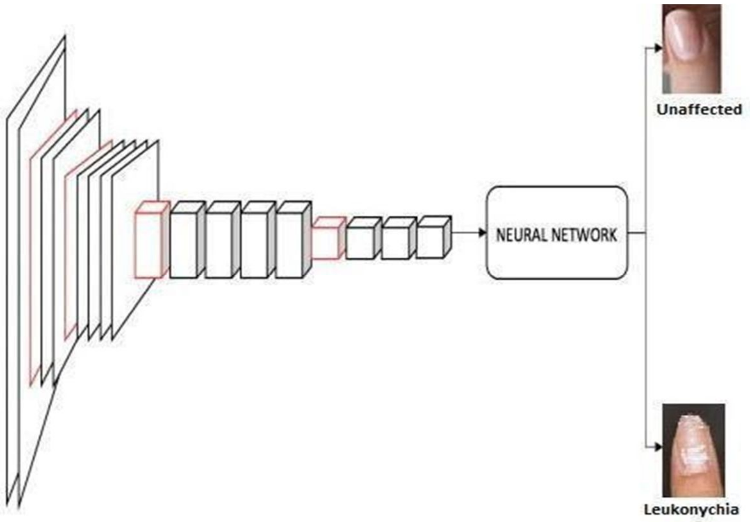

Fig. 3. Our disputed Neural Network Skeleton had 19 layers (inception-v3) which were sent to an ANN to distinguish leukonychia affected nails.

We figure out the Receiver Operating Characteristic curve for calculation of accomplishment of the skeleton. It evolves by sketching "Specicity" (X-coordinate value) and "Sensitivity" (Y-coordinate value) that gives an output in a graph. The value of Region Inside Curve which is nearer to 1, denotes more degree of superiority [8]. Furthermore, we used various models of Logistic Regression (LR), Artificial Neural Network (ANN), Support Vector Machine (SVM), Naive Bayes (NB), Adaboost, CN2 Rule Induction, kNN and Random Forest (RF) on this dataset to get better accuracy evaluation (Fig. 4).

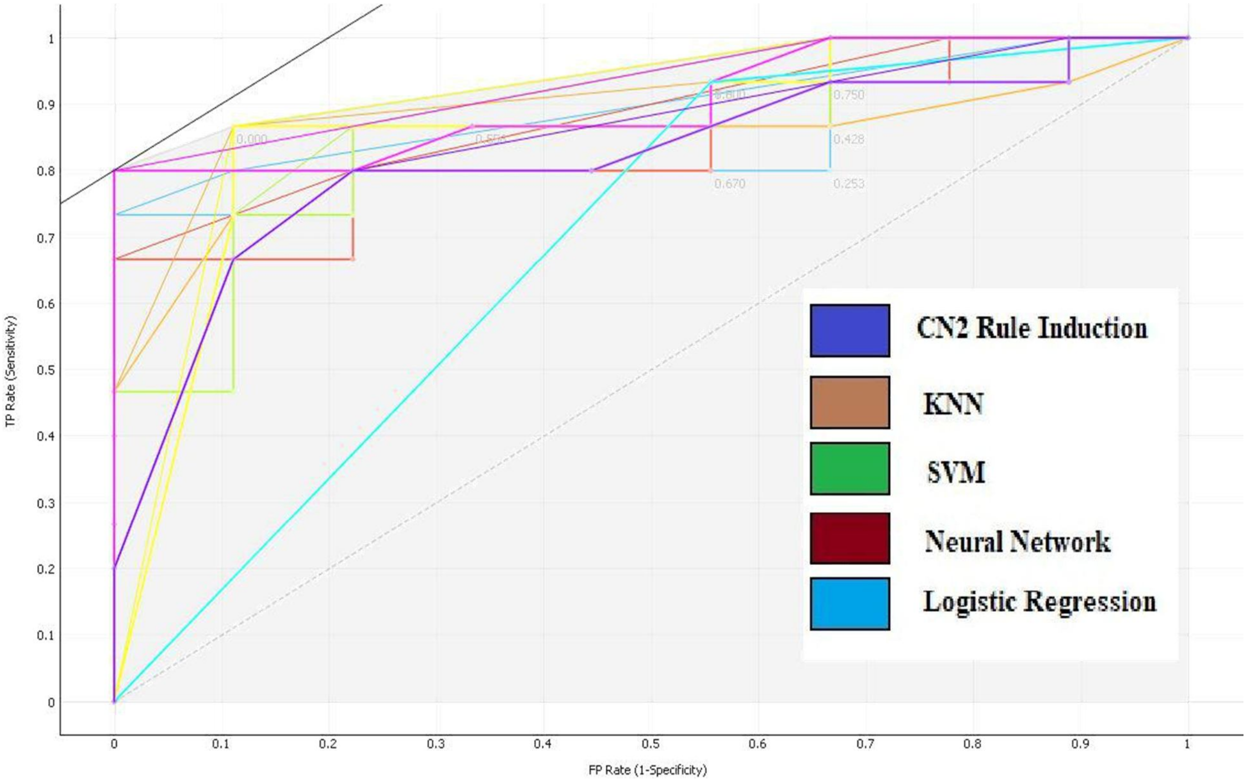

Fig 4. ROC Curve

VI. RESULTS AND DISCUSSIONS

All the three perspectives we have taken care for achieving the correct result. Now, in this section we come up with the final results achieved by the tests done in the above sections .We can see that our process of detecting Leukonychia is best 93.8% when compared with other latest models .we can see that result coming from our CNN is in comparison with the practicing expert dermatologist and latest machine learning algorithms with very few preprocessing. Our results have come up after using pre trained models on different datasets. We have looked upon three scenarios. In the 1st scenario, a 16-convolutional layer, 5 max pooling layers are applied to all hidden layers. After this we have applied all 3 properly associated layers, RELU activation is then projected to all hidden layers. We obtained the best accurate result of 84.0% with Logistic Regression, AdaBoost and KNN. In the 2nd scenario, after using 42 layers of DNN of low rate of errors, that made us achieve a better result of 90.6% accuracy with Random Forest, ANN, LR and Naive Bayes. It is seen that the result obtained is much better than VGG Net. In the 3rd scenario , a 19 CL deep model that is used achieved accuracy of is 93.8% with CN2 Rule Induction, 87.5% with K–NN, 84.4% with Support Vector Machines (SVM) [1].After all we can see after assessing our output the change in the accuracy of all the synopsis are tantamount, and during our training time occasions are mostly almost equal but in scenario of training a CN2 Rule Induction model, thus bringing the most favourable way for proposing scenario 3 (93.8%).we can look that after increasing the convolutional layers in framework, our accuracy increased because of efficient extractions of feature (Table 2).

Table 2. Accuracy Assessment

Feature extractor	Models	AUC	Accuracy	Precision
VGG-16	SVM	0.881	0.760	0.743
	KNN	0.937	0.840	0.862
	Naïve Bayes	0.873	0.800	0.840
	AdaBoost	0.802	0.840	0.840
	Neural Network	0.921	0.800	0.810
	Random Forest	0.841	0.800	0.793
	Logistic Regression	0.913	0.840	0.836
	CN2 Rule Induction	0.627	0.680	0.680
VGG-19	SVM	0.967	0.812	0.856
	KNN	0.985	0.875	0.896
	Naïve Bayes	0.952	0.906	0.909
	AdaBoost	0.800	0.812	0.812
	Neural Network	0.887	0.906	0.909
	Random Forest	0.958	0.906	0.918
	Logistic Regression	0.988	0.906	0.918
	CN2 Rule Induction	0.742	0.750	0.750
Inception -- V3	SVM	0.936	0.844	0.845
	KNN	0.964	0.875	0.875
	Naïve Bayes	0.841	0.781	0.807
	AdaBoost	0.705	0.781	0.773
	Neural Network	0.914	0.812	0.816
	Random Forest	0.875	0.812	0.807
	Logistic Regression	0.895	0.812	0.816
	CN2 Rule Induction	0.989	0.938	0.943

VII. CONCLUSION

In Our methodology we used Deep Convolutional Neural Network to predict affected nails (Leukonychia) and unaffected nails that are predicted from data images obtained result of 93.8% correctness. Because of the scarcity of an already present standard set of data having a large quantity of image dataset of leukonychia, this exercise accumulated a lot of problems and complications, as we know there is still some opportunity for continuation in the nearest future. We are looking to develop better accurate and specific methods used in computation of probability of hair problems, estimating skin problems, and a lot of different nail diseases in the future. Furthermore, this algorithm can help skin experts to figure out client’s health problems that don't require much manual work.

REFERENCES

[1] Shadab, Malik & Dwivedi, Mahavir & S N, Omkar & Javed, Tahir & Bakey, Abdul & Raqib, Mohammad & Chakravarthy, Akshay. (2019). Disease Recognition in Sugarcane Crop Using Deep Learning. 10.13140/RG.2.2.21849.47209.

[2] 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.

[3] Nijhawan, Rahul & Das, Josodhir. (2020). A Proposed Framework Approach for Mapping Glacier Hazard Zones. 10.1007/978-981-13-7067-0_44.

[4] Nijhawan, Rahul & Jindal, Radhika & Sharma, Himanshu & Raman, Balasubramanian & Das, Josodhir. (2019). A Deep Learning Framework Approach for Urban Area Classification Using Remote Sensing Data. 10.1007/978-981-32-9088-4_37.

[5] Rawat, Sarthak & Bisht, Abhishek & Nijhawan, Rahul. (2019). A Classifier Approach using Deep Learning for Human Activity Recognition. 486-490. 10.1109/ICIIP47207.2019.8985926.

[6] Aishwarya, & Goel, Akansha & Nijhawan, Rahul. (2019). A Deep Learning Approach for Classification of Onychomycosis Nail Disease. 10.1007/978-3-030-30577-2_98.

[7] A. Bhalla, A. Ahuja, P. Pant and A. Mittal, "A Multimodal Approach for Automatic Cricket Video Summarization," 2019 6th International Conference on Signal Processing and Integrated Networks (SPIN), Noida, India, 2019, pp. 146-150.

[8] Nijhawan, Rahul & Joshi, Deepankar & Narang, Naman & Mittal, Aditya & Mittal, Ankush. (2019). A Futuristic Deep Learning Framework Approach for Land Use-Land Cover Classification Using Remote Sensing Imagery. 10.1007/978-981-13-0680-8_9.

[9] Varshni, Dimpy & Thakral, Kartik & Agarwal, Lucky & Nijhawan, Rahul & Mittal, Ankush. (2019). Pneumonia Detection Using CNN based Feature Extraction. 1-7. 10.1109/ICECCT.2019.8869364.

[10] Nijhawan, Rahul & Rishi, Megha & Tiwari, Amit & Dua, Rajat. (2019). A Novel Deep Learning Framework Approach for Natural Calamities Detection: Proceedings of Third International Conference on ICTCS 2017. 10.1007/978-981-13-0586-3_55.

[11] S. Kumar, A. Tyagi, T. Sahu, P. Shukla and A. Mittal, "Indian Art Form Recognition Using Convolutional Neural Networks," 2018 5th International Conference on Signal Processing and Integrated Networks (SPIN), Noida, 2018, pp. 800-804.

[12] P. Shukla, T. Gupta, A. Saini, P. Singh and R. Balasubramanian, "A Deep Learning Frame-Work for Recognizing Developmental Disorders," 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), Santa Rosa, CA, 2017, pp. 705-714.

[13] I. Dua, P. Shukla and A. Mittal, "A vision based human - elephant collision detection system," 2015 Third International Conference on Image Information Processing (ICIIP), Waknaghat, 2015, pp. 225-229.

[14] Tie-Fei Liu, Wing-Kin Sung and A. Mittal, "Learning multi-time delay gene network using Bayesian network framework," 16th IEEE International Conference on Tools with Artificial Intelligence, Boca Raton, FL, USA, 2004, pp. 640-645.

[15] N. Mamgain, E. Mehta, A. Mittal and G. Bhatt, "Sentiment analysis of top colleges in India using Twitter data," 2016 International Conference on Computational Techniques in Information and Communication Technologies (ICCTICT), New Delhi, 2016, pp. 525-530.

[16] S. N. Chandran, D. Gangodkar and A. Mittal, "Parallel implementation of local derivative pattern algorithm for fast image retrieval," International Conference on Computing, Communication & Automation, Noida, 2015, pp. 1132-1137.