Survey on the use of CNN and Deep Learning in Image Classification

Siddhant Dani, Prof. P. S. Hanwate, Hrishikesh Panse, Kshitij Chaudhari, Shruti Kotwal

Computer Engineering Department,

NBN Sinhgad School of Engineering, Pune-411041, India.

Abstract : With an increase in the speed of data generation , the widespread use of cameras for automation and surveillance, and the need for visual feedback for artificially intelligent devices around the world, the mass of image data being produced today has increased rapidly. The need for efficient image processing has risen to simplify the image related tasks. Categorical Image Classification requires thousands of images to train the system. Also the system requires a large amount of time to extract the features and for classification as well. Hence, for image recognition tasks, deep convolutional neural networks have been introduced and have obtained promising results in recent years. They are being widely deployed to analyze, detect and classify images for a diverse range of tasks. This paper offers a summary of existing systems designed for image classification using CNN. At first, it defines the terms Image Classification and CNN and reviews current surveys dealing with CNN implementations for Image Classification. Then, included a general overview of What is Image Classification?, and CNN architecture. Moreover, This paper provides a taxonomy of state-of-the-art methods for efficient and accurate CNN applications in relation to the Image Classification problem. Based on the present study, we stress the open barriers of research and potential future work directions in CNN based Image Classification systems.

IndexTerms - Image Classification, CNN, Deep Learning, Image Processing, Deep Networks, Neural Networks.

I. INTRODUCTION

Image processing has been defined by Lillsand and Kiefer as involving the use of computers to manipulate digital images. It is a vast topic and generally involves processes that are very complex mathematically. The aim of creating an image archive is to manage and easily retrieve the large image data that gets collected after different events. By classifying the image data into different categories based on domain awareness, this task could be resolved. Some simple operations are involved in image processing, such as image restoration/rectification, image enhancement, image classification, fusion of images, etc. The classification of images forms an essential part of processing images. Automatic allocation of images to thematic groups is the purpose of image classification. Supervised classification and unsupervised classification are two types of classifications.

The method of image classification requires two stages, followed by testing and training of the system. The training process involves taking the image's characteristic features (forming a class) and forming a special definition for a specific class. Depending on the form of classification problem, the process is carried out for all classes: binary classification or multi-class classification. The testing phase involves classifying the test images into different groups for which the system has been trained. This class assignment is carried out on the basis of the partitioning of the training features between classes.

For classification, Machine Learning systems may be used to extract features that represent the semantic content of the image data. It is possible to categorize algorithms based on a learning process or using Functional Similarities. Deep learning algorithms focused on functional similarities are Neural Network-enthusiastic algorithms. Deep learning algorithms are approaches inspired by neural networks that use massive data sets to predict the outcome in a semi-supervised learning environment.

The concept of deep learning was introduced in 2006 at first. And since then it is still developing. Deep learning is characterized as a class of techniques for machine learning that exploit several layers of non-linear processing of information for supervised or unsupervised extraction and transformation of features and for analysis and classification of patterns. For implementation of CNN a large volume of data is needed in order to train the model.

The paper is structured as follows; Section II gives information about the studies related to our topic. Section III gives us brief information about the concept of Image Classification , Deep Learning and CNN. While in Section IV we discuss the implementations of CNN for Image Classification. Section V discusses challenges and future work and in section VI we conclude the paper.

II. OVERVIEW AND RELATED WORK

We found 10 studies that reviewed the implementation of CNN for Image Classification. Fig. 1 lists out the names of authors of the study papers, their proposed systems, advantages and disadvantages of their proposed systems.

Two of the papers discussed using CNN and SVM to solve the image classification problem. One paper focussed on the use of Capsnet to classify images. A further two papers discussed the development of the CNN-based feature extraction model and then use of SVM to classify the images. One paper focussed on the use of Bayesian theory and generalized Lloyd algorithm for image classification. And two papers proposed the use of CNN and TensorFlow for solving image classification problems. Also one paper focussed on using RESNET for image classification and another paper proposed only using a small CNN to classify images accurately and successfully.

Thus Fig. 1 focuses on the major contributions of the previous studies conducted regarding the use of CNN and Deep Learning approaches for image classification. We can say that the below-mentioned papers have built a strong base for CNN based image classification systems, but our survey focuses on the following points:

- We review the need and use of Image Classification and CNN.
- We review existing research on the CNN based Image Classification systems.
- We discuss the challenges and potential future directions for research in the CNN based Image Classification area.

Year of Publication	Author and Paper Name	Objective/Description	Proposed System/Methods	Advantages	Disadvantages/Challenges
2001	Aditya Vailaya, Mario A. T. Figueiredo, Anil K. Jain, Hong-Jiang Zhang, "Image Classification for Content-Based Indexing"	The authors discussed the need of content based image classification and their approach to solve the problem of inefficient image classification	Image classification using Bayesian theory. The required probability density functions are estimated by using vector quantization. A generalized Lloyd algorithm is used to obtain VQ suitable for classification.	Small no of codebook vectors to represent each class. Allows integration of multiple features. Degree of Confidence used to add reject option into classifiers.	Different images may have same features hence difficult to classify images. Introducing a Reject option is difficult. Slower than most modern methods.
2020	Sai Yeshwanth Chagarti, Ipseeta Nanda, Koteswara Rao Pandi, Niraj Kumar, "Image Classification using SVM and CNN"	To classify images in different categories using Support Vector Machine and CNN	The model described in this paper classifies images using SVM and CNN. For both implementations images are converted to array of numbers using NumPy library functions. And then SVM and CNN are used to separately to classify images.	Achieved 82% accuracy using SVM. Comparison of SVM and CNN performance on same dataset. Achieved 83.57% accuracy using CNN.	Achieveing higher accuracy by using SVM is difficult. Drawbacks of SVM in data-rich environment can be clearly seen. Trying to use CNN and SVM in same architecture.
2019	Turan Gokhag Altundogan, Mehmet Karakose, "Image Processing and Deep Neural Image Classification Based Physical Feature Determiner for Traffic Stakeholders"	To determine traffic stakeholder type by using Image Processing and Deep Learning. To determine the size feature and color of the traffic stakeholder.	In this method video frames are used to detect traffic stakeholders contour and then images are cropped according to contours. These cropped images are then sent to deep image classifier. Then vehicle size is calculated in pixels and RGB values are converted to HSV values in order to determine the vehicle color.	Classification is done by achieving more than 90% accuracy for all classes. Use of TensorFlow to create Deep learning model. Use of Java and OpenCV to create a Physical Model.	For some stakeholders color or size values cannot be determined. Image preprocessing takes more time than classification. To develop a fully working web service for this model.
2019	Zhiyong Dong, Sheng Lin, "Research on image classification based on Capsnet"	To use Capsnet to classify images of MNIST and CIFAR10 datasets. To compare performance of Capsnet and CNN.	The Capsnet model described has 2 convolutional layers and one fully connected layer. Conv1 has 256, 9*9 kernels and step size of 1. This layer converts intensity of pixels to be used as a local feature. The second layer, PrimaryCaps is a 32-channel capsule layer. The PrimaryCaps layer has total 32*6*6 outputs. DigitCaps layer has a 16D capsule for each digital class that accepts input.	Accuracy of Capsnet for MNIST dataset is 99.71%. Accuracy of Capsnet for CIFAR10 dataset is 75.30%. No of parameters used by Capsnet is about 1/4 of the CNN. Uses dynamic routing.	For dataset with complex images Capsnet performance is poor. Poor performance of CNN against Capsnet. CNN uses 4 times more parameters than Capsnet to classify images.
2019	Mrs. Arpana Mahajan, Dr. Sanjay Chaudhary, "Categorical Image Classification Based On Representational Deep Network (RESNET)"	To extract image features from a pre-trained Representational deep Neural Network and use those features to train SVM inorder to classify thousands images into 8 categories.	An eighteen layer and or 34 layer ResNET model is used to extract image features from 224*224*3 dimensional images stored in database. Then SVM is used to predict future data using regression. SVM also performs mapping of low dimensional space into high dimensional space. SVM uses linear classifiers to separate the data.	18 layer RESNET has the highest accuracy of 93.57%. RESNET makes feature extraction fastest and easier. Faster than many conventional CNN's.	Use of more layers in RESNET reduces its performance. SVM is one of the most powerful but very old technique.
2019	Shyava Tripathi, Rishi Kumar, "Image Classification using small Convolutional Neural Network"	To implement image recognition using a small CNN which proposes less complexity and yielding good accuracy for multiple datasets.	The model contains a total of 13 layers. The convolutional layers are connected with input images during training. This model is retained during training and is utilized to operate on input image dataset to predict whether the given image belongs to the classes known to the model.	On 100th iteration model achieves accuracy of 99%. 98% validation accuracy. After 100th iteration 0.12 validation score.	Since model proposed is applied to dataset only once no. of iterations to get max accuracy is very high. At initial iterations accuracy is very low & validation loss is very high.
2019	Yanyan Wang, Chengyong Wang, Lenghan Luo, Zhigang Zhou, "Image Classification Based on transfer Learning of Convolutional neural network"	By using an algorithm, transfer learning which is based on CNN, combined with image histogram of oriented gradient feature extraction method and SVM, classify images into respective categories.	The HOG features of the training sample similar to the attributes of the samples to be classified are extracted, then the hog features of the training samples are imported into the SVM classifier to get the pre-classification results. The pre-classification results are used as training samples to train the transfer network of CNN for getting new learning model, this model can be used to classify similar preclassification samples.	The model achieves an accuracy of 95%. Reduces training time by a huge amount i.e. from 8 hours to 2 minutes. Avoides the problem of over-fitting caused by too small dataset.	Generalization ability of transfer model is poor in new situations. Problem of negative transfer and under-adaptation needs to be solved before using in actual situations.
2018	Santosadha Panigrahi, Anupa Nanda, Tripti Swarnkar, "Deep Learning approach for Image Classification"	To classify images into either dogs/cats category using deep learning frameworks such as CNN, Theano, Keras & TF.	The HOG features of the training sample similar to the attributes of the samples which to be classified are extracted, then the hog features of the training samples are imported into the SVM classifier to get the pre-classification results. Finally, the pre-classification results are used as training samples to train the transfer network of CNN for getting new transfer learning model, this model can be used to classify	Highest accuracy of 84.45% for 30 iterations. Theano is used for rapid mathematical calculations. Three layer architecture of CNN.	More no. of misclassified images for early iterations. Large dataset and more no. of layers can improve the accuracy of the model.
2017	Ardhendu Bandhu, Sanjiban Sekhar Roy, "Classifying multi-category images using Deep Learning : A Convolutional Neural Network Model"	This paper introduces an image classification model using a convolutional neural network and tensor flow.	Image classification using CNN and Tensor Flow. Tensor flow defines computations as graphs and these are made with operations. Image data is fed to a tensor flow which allows us to shape the input data as we want. Then input features are extracted and converted into a input tensor. By using Input Tensor the model can be trained to make predictions.	High Accuracy. Easy to Understand. CPU based system.	Time required to train is very high. No. of iterations required to desired accuracy are very high. Complex Design.
2018	Md Tohidul Islam, S.M. Nafiz Karim Siddique, Sagidur Rahman, Tasfeed Jabid, "Image Recognition with Deep Learning"	To Classify food images using deep learning methdologies CNN, SVM, etc.	The model described uses a Convolutional Neural Network. Image Pre-processing techniques such as random rotation and horizontal flips are used to neglect exact positions of items in image. The CNN V3 model has 7 layers as follows: Convolutional layer, Max pooling layer, Average pooling layer, Concat layer, Dropout layer, Full connected layer, Softmax layer.	V3 CNN model has accuracy of 92.86%. layered design of CNN. Use of separate datasets for training, testing and validation purposes.	A significant gap between training and testing accuracy. Other models such as RNN, DCNN can also be applied. Performance can be improved by using feature based models which take less computational time. Larger dataset with more categories can be used.

Fig. 1: Survey Table

III. IMAGE CLASSIFICATION AND CONVOLUTIONAL NEURAL NETWORKS(CNN)

A. Image Classification

- **What is Image Classification?**

The task of identifying what an image represents is called as Image Classification. The objective of the classification process is to categorize all pixels into one of many groups or "themes" in a digital image. It is then possible to use this categorized data to create thematic maps of the objects present in an image. Multispectral data is typically used to execute the classification, and the spectral pattern within the data for each pixel is actually used as the numerical basis for the categorization. The purpose of image classification is to define and depict the characteristics that appear in an image as a distinct gray level (or color) in terms of the object these characteristics reflect. Image classification is the most important part of image analysis or image processing. The two main types of classifications are Supervised classification and Unsupervised classification.

- **Supervised Learning**

In supervised learning we classify or identify the classes to which an image belongs depending on the information present in an image. These are called "training sites". A mathematical characterization of the reflectance for each class is then established using the image processing software framework. This process is also referred as "signature analysis" and can entail the creation of a characterization as basic as the mean or the reflectance range on each band or as complex as comprehensive mean, variance and covariance analysis over all bands. When a statistical classification for each information class has been achieved, the image is then characterized by analyzing the reflectance for each pixel and making a determination about which of the signatures it most resembles.

- **Unsupervised Learning**

Unsupervised classification is a methodology that analyses a vast number of undefined pixels and separates them into a number of classes depending on the image values of natural groupings. Unsupervised classification does not include analyst-specified training data, like supervised classification. The basic principle is that values should be close together in the measurement space within a given cover type (i.e. have identical gray levels), while data should be comparatively well differentiated between different types (i.e. have very different gray levels).

In order to determine the identities and information values of the spectral classes, classes arising from unsupervised classification are spectral categorized based on normal groupings of image values, the identity of the spectral class is not initially known, and classified data must be correlated with any of the reference data (such as larger-scale imagery, maps, or site visits).

- **Image Classification Techniques**

1. Neural Networks.
2. Support Vector Machine classifier(SVM).
3. Genetic Algorithms(GA).

- **How does it work?**

In the training phase, an image classification model is provided with images along with their associated labels. Each of these labels belong to a class or concept name which the model will learn to identify.

After giving sufficient amounts of images as training data (usually thousands of images per class), the model built will be able to predict whether the new images belong to any of the classes on which it has been trained on.

Whenever we provide a new image as input to the model, it will output the probabilities for the image representing the types of classes it was trained on. Each of these probability values will refer to a class label. Depending on the highest probability value, we can determine the class of the image.

B. Convolutional Neural Networks

- **What is a Convolutional Neural Network?**

A Convolutional Neural Network is a deep learning algorithm that can take an input image, assign value to different aspects/objects in the image (learnable weights and biases) and be able to distinguish one from the other.

In deep learning, CNN's are most commonly used for visual imagery. They have various applications such as image and video recognition, suggestion systems, image detection, medical image interpretation, processing of natural language, brain-computer interfaces, and financial time series.

In a regular neural network there are three types of layers, an input layer in which we give input to our model. A hidden layer, which accepts the input from the input layer. There can be many hidden layers depending on the model and data size. An output layer, which accepts the output of a hidden layer and converts the output of each class into a probability score for each class**.**

- **CNN Architecture**

CNN's are neural networks that share their parameters. It is a sequence of layers and every layer transforms one volume to another through a differential function.

Different types of layers used to build CNN's are as follows:

1. Input Layer: This layer holds the raw input of an image with parameters such as height, width and depth.
2. Convolution Layer: This layer computes the output volume by computing dot product between all filters and image patches. If we use N filters then we will get output volume as W*H*N, where W is width of image, H is height of image.
3. Activation Function Layer: The activation function of this layer is applied to the output of the convolution layer. RELU: max(0, x), Sigmoid: 1/(1+e^-x), etc., are some common activation functions. The volume stays unchanged, so the volume of output would be the same as the convolution layer.
4. Pool Layer: This layer is placed regularly in the CNN's and its key function is to decrease the volume size, which reduces the memory usage by making computations fast and also avoids overfitting. Max pooling and average pooling are two generic forms of pooling layers.
5. Fully-Connected Layer: This layer is a standard layer of the neural network that takes data from the previous layer and determines the score of the class and returns the 1-D array of size relative to the number of classes.

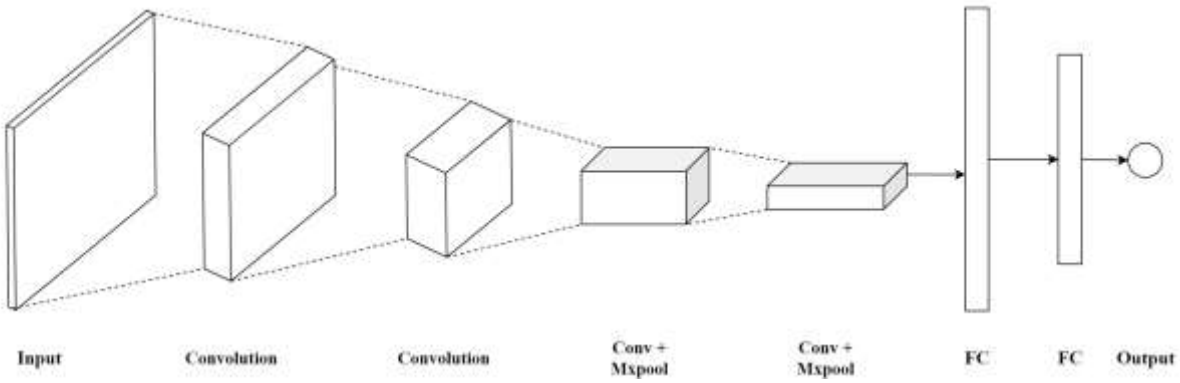

Fig. 2: Typical CNN Architecture

Advantages:

- Very High accuracy in image recognition problems.
- Automatically detects the important features without any human supervision.
- Weight sharing

IV. CNN-BASED IMAGE CLASSIFICATION MODEL

In this section we review some of the existing systems that have been designed using the concepts of CNN and Image Classification that we have discussed in the previous section.

One of the many approaches used to classify images, in [1] the authors discussed using both CNN and SVM to classify images separately and comparing the results to determine which method is superior. The authors used a customized dataset containing over 350 images from 5 classes. For classifying images using SVM the dataset used was enough, but for CNN based approach the dataset used didn't give desired results. According to the authors they were able to achieve a successful classification accuracy of 93% for the given dataset using SVM classifier. In order to use CNN for classification the authors used some data augmentation techniques to convert each image from the dataset into 7-8 similar images as CNN requires a large dataset for training. The newly formed dataset contained over 3000 images. The different data augmentation techniques used by the authors were changing color space from RGB to CMY, dithering of the image, random translation of the image and flipping the image vertically or horizontally. The new dataset with over 3000 images was loaded into python by using a user defined function which converted the images into 64 by 64 and then flattened all the images using NumPy library functions resulting in an array of numbers for each image in the dataset. This number array can be easily used by any classification method such as CNN, SVM, etc for easier classification.

As the authors of [1], already determined the accuracy of 82% using SVM for the newly generated dataset, they used a LeNet5 architecture similar to CNN for classification. In LeNet5 the input layer converts a 120*120*1 image into a 116*116*6. The output of the input layer is then passed through the max pooling layer and is reduced to 58*58*6. Again this output is passed through filter and max pooling layer and is reduced to 27*27*16. The result is then flattened until a fully connected layer of 84 neurons is obtained. At last, a soft-max function is applied to get the result in terms of the 5 classes. The authors of [1] were able to achieve an accuracy of 93.57% using the CNN based LeNet model for the newly generated dataset. Thus they were able to compare the results of both SVM and CNN and were able to determine the superiority of CNN over SVM.

Classes	Precision	Recall	F1 Score
Dalmatian	0.78	0.82	0.8
Dollar Ball	0.88	0.93	0.9
Pizza	1	0.88	0.93
Soccer Ball	0.88	0.7	0.78
Sunflower	0.83	1	0.91
Micro-Average	0.86	0.86	0.86

Fig.3 : SVM Results

	Dalmatian	Dollar Ball	Pizza	Soccer Ball	Sunflower
Precision	0.94	1	0.83	0.97	0.91
Recall	0.86	0.94	0.96	0.91	0.93
F1-Score	0.89	0.97	0.89	0.94	0.92

Fig. 4 : CNN Results

In another study[2], the authors wanted to determine physical features of vehicles using image processing and deep learning. The goals set by the authors were to identify the type of traffic stakeholder, to determine size features of stakeholder and to determine the color of the traffic stakeholder if they are not a pedestrian. The data used for learning was video frames from the traffic surveillance system. The authors of [2] used gray conversion and background subtraction for preprocessing the video frames.

Background subtraction helps in finding contours efficiently. After background subtraction the images were obtained in binary format. Contours are used in image processing for finding most relevant pixel groups. After finding the contours these images are cropped if the contour region is too big in size. Then these cropped images are sent to the Deep Neural Classifier model. The model is trained on these values of pixels obtained in the preprocessing stage to determine the type of traffic stakeholder. The authors used HSV conversion method to identify the color of the traffic stakeholder. The HSV values were calculated for all the pixels in cropped images and were again fed to the Deep Neural Classifier to determine the color classes.

The model was built using TensorFlow and trained offline on thousands of images belonging to each class. The deep neural classifier classifies the type of the traffic stakeholder in seven classes. They are as follows Vehicle, Truck, Motorcycle, Pedestrian, Nonstakeholder, Bus and Trailer. The authors of [2], tested their Deep Neural Classifier model by using more than 2000 images belonging to each class. They obtained accuracy of more than 90% for all the classes. But they also discovered that the image preprocessing task takes more time than the actual classification task. And even after image preprocessing the model was not able to determine color and size values for some images.

The authors of [3], used another approach to classify images which involved the use of Capsnet to measure its performance on the MNIST and CIFAR10 datasets to resolve the classification problem. Capsnet is a new and very innovative deep learning method. Capsnet is very similar to CNN in terms of architecture design. The main difference between CNN and Capsnet is that Capsnet embeds neural layers into one another while CNN keeps adding new layers to the network to make a deep network.

The MNIST dataset has around 70000 images belonging to 10 classes. It contains 60000 training images and 10000 images in testing dataset. The authors of [3], described the Capsnet architecture for MNIST dataset as having 3 layers, 2 convolutional layers and one fully connected layer. The first layer Conv1 has 256, 9*9 kernels and Relu as activation function. This layer converts the intensity of pixels to be used as a local feature. The second layer PrimaryCaps is a 32 channel convolutional layer. It has a total of 32*6*6 outputs. The third layer DigitCaps has a 16D capsule in each class each capsule accepts input from other capsules below. The authors upon using this model on MNIST dataset were able to achieve an accuracy of 99.71%. The number of parameters used by the Capsnet were 8.5M.

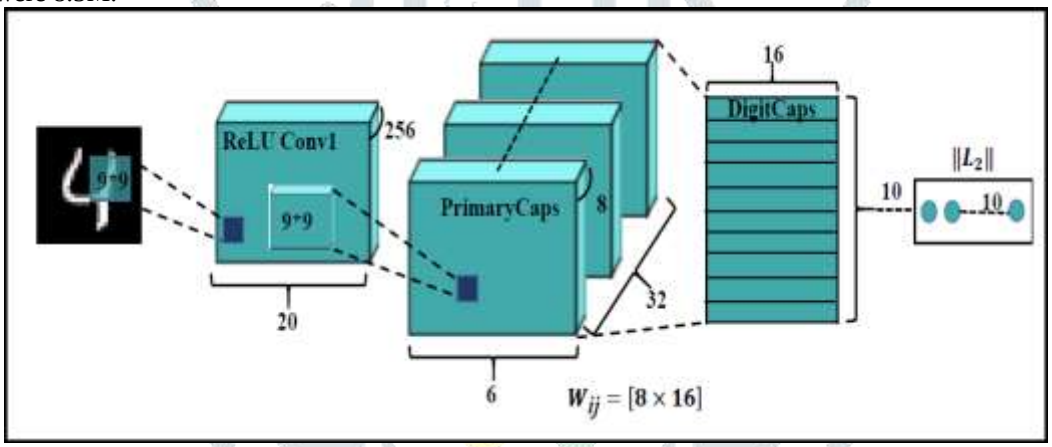

Fig.5 : Capsnet Architecture

Similarly, the CIFAR10 dataset has about 60000 images of size 32*32 belonging to 10 classes. The architecture design used for CIFAR10 dataset is the same as the architecture for MNIST dataset. The authors were able to achieve an accuracy of 73.30% for this dataset.

The authors of [3] also determined that with just a little modification to the architecture of CNN, the newly obtained Capsnet model gives better performance. But the performance of the Capsnet is poor for a dataset with complex images.

In [4], the authors described a categorical image classification system using Representational Deep Network(RESNET). They described a system which uses RESNET for feature extraction and based on the output of the RESNET they used SVM to classify images into different classes. A RESNET is a deep convolutional neural network which has more layers than a typical CNN. The model was trained on an eight population order dataset which has about 2698 images of size 224 by 224 by 3. In the RESNET the arrangement of layers constructs a hierarchical representation of the images. The deeper layers accommodate the high level features from the image. The extracted features are pooled over all spatial locations by the global pooling layer. The authors described two RESNETs having 18 and 34 layers respectively.

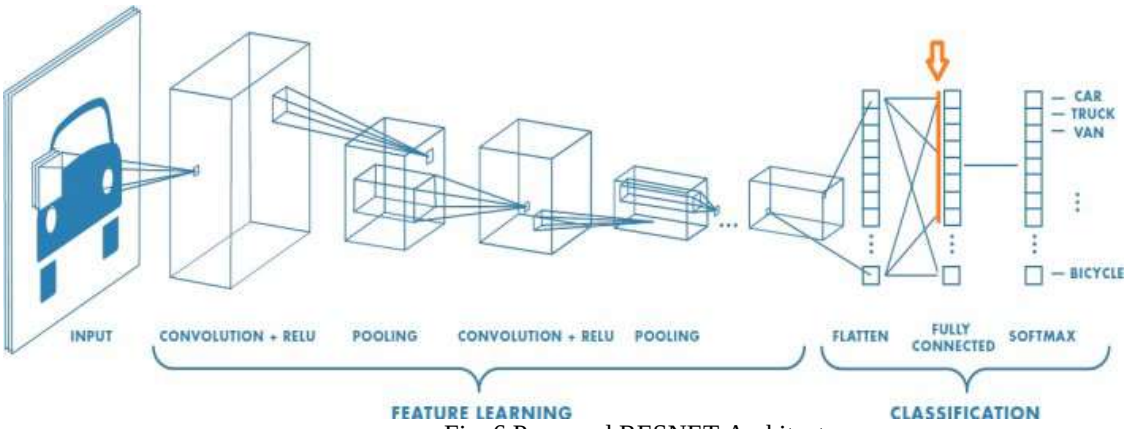

Fig. 6 Proposed RESNET Architecture

The features extracted from the RESNET are then passed to the SVM classifier. SVM is a supervised learning binary classifier. The authors used SVM as a classifier for determining the output class of the testing images. The 18 layer RESNET used for feature extraction along with SVM was able to achieve an accuracy of 93.57%. While the 34 layer RESNET along with SVM achieved an accuracy of 91.58%.

The authors of [4], were able to depict that the RESNET is faster than any other conventional CNN. And it makes feature extraction faster and easier. But they also discovered that use of more layers in RESNET reduces its performance.

Another approach proposed by authors of [5] was the use of a small & simple convolutional neural network for image recognition. They used a dataset of 100000 images belonging to 200 classes. They divided the dataset in three parts: training dataset, testing dataset and validation dataset to avoid overfitting.

The model described in [5] will use the training dataset for training purposes and then with the help of testing dataset it will identify and separate images according to their classes. The model is made up of 13 layers namely, 5 convolution layers, 3 max pooling layers, 1 dropout layer, 1 flattening layer, 2 fully connected layers and 1 softmax layer. The input images are 64*64 in size with 3 color channels which is given as input to the first convolutional layer. This layer is followed by a max pool layer. The first convolutional layer has of 32 filters of size 3*3 and gives the output in form of activation map of 64*64*32 which is passed to the max pool layer. The max pool layer has a stride of 2,2 which reduces the dimensions of the image by half. Then the authors used a ReLU activation function to introduce non-linearity in the model.

Similarly this process is carried out for other 5 convolution layers, the output of the previous layer is fed to the next convolution layer and so on. But in the fifth convolution layer the authors increased the number of filters to 64 which produces the output of size 8*8*64. The output of this layer is then fed to a dropout layer with dropout rate of 0.8. The function of the dropout layer is to drop neurons that are selected randomly during training.

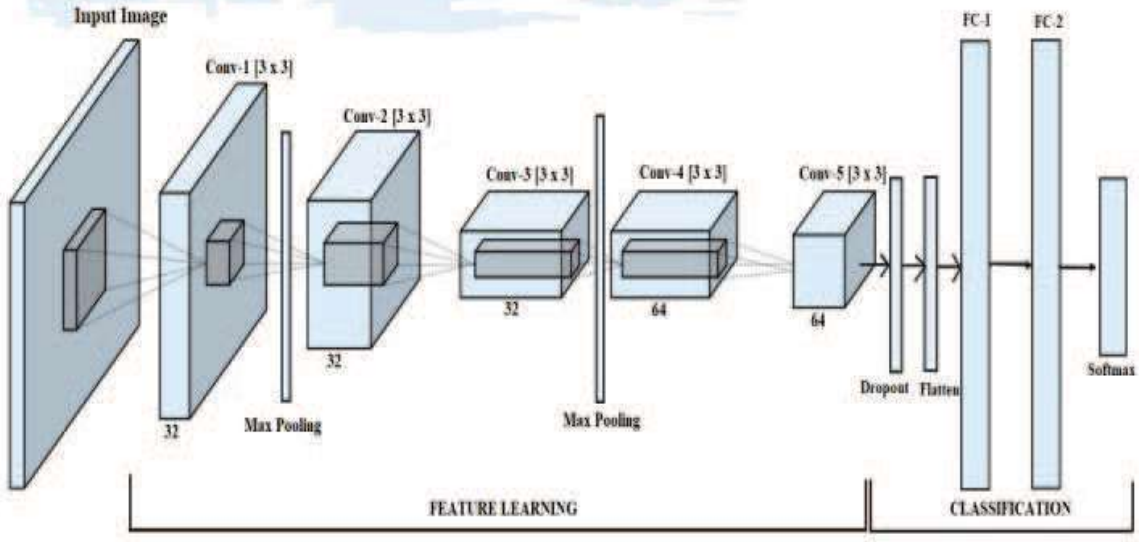

Fig. 7: Architecture of the Proposed 13 layer- CNN

The output of the dropout layer is then passed to the flattening layer to convert the multi ranked tensor into a 1 rank tensor. This tensor is then passed to the first fully connected layer which has 128 neurons. A ReLU function is applied to this tensor to obtain the output of the first fully connected layer. The output obtained is then passed to the second fully connected layer and a similar process as the first fully connected layer is carried out. Then the output obtained from the second fully connected layer is passed to a softmax classifier which outputs the class of the input image.

The authors of [5], ran their model for about 100 epochs. By the 30th epoch the model was able to achieve a 50% training accuracy, and 45% validation accuracy. On the 100th epoch the model was able to achieve a 99% of training accuracy and 96% of validation accuracy which is a great result. The main advantage of this model was that it was able to predict outputs for 200 image classes. And also it achieved a great accuracy with just 13 layers. But at the same time as the model is only applied to the dataset once, the accuracy for training and validation is very less in the initial iterations. Hence the training process needs to be carried out multiple times which just increases the time required.

The system proposed in [6], uses a HOG+SVM and Alexnet transfer learning algorithm to solve the problem of successful image classification. It also focuses on solving the problem of under-adaptation in deep learning. Alexnet is a type of CNN which uses transfer learning algorithms to train the model. HOG is also known as Histogram of Oriented Gradient.

The HOG feature used in [6], is used to describe the shape and appearance of an image by the density of the gradient and direction of the edge. It is used as a feature extraction method. In HOG, the RGB images are converted to grayscale images and are used as input images and then they are normalized using the Gamma correction method. Then a set of formulas are used to calculate the gradient of each pixel. Next, these images are divided into 6*6 cells, and the gradient histogram for each cell is calculated. Then by using these cell feature vectors, HOG characteristics of a block are obtained. Then the HOG features from all the cells are combined to form a final feature vector, to be used for classification. Then this final feature vector is passed to the SVM classifier to obtain the pre-classification results.

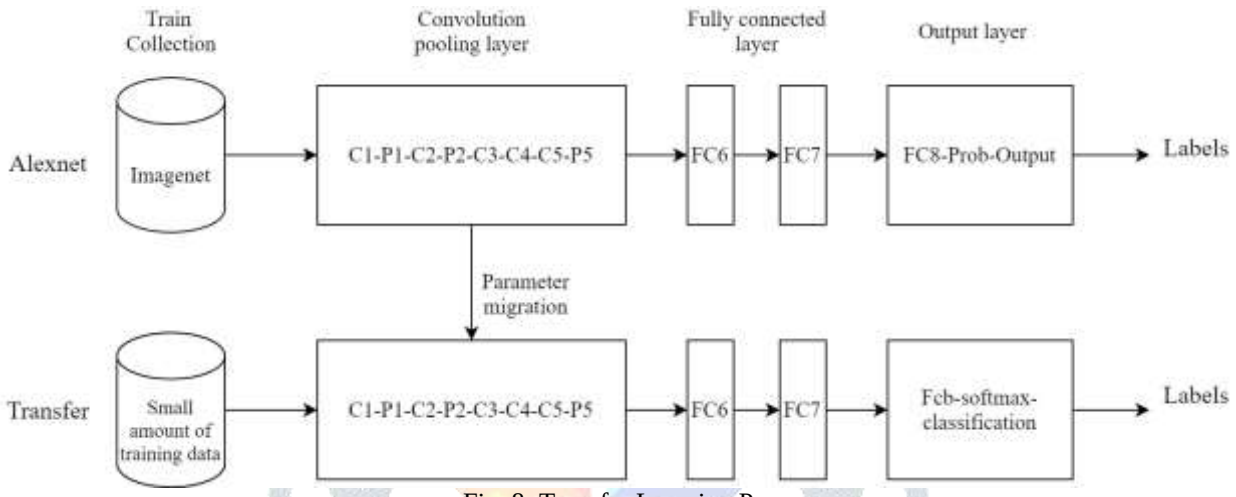

Fig. 8: Transfer Learning Process

An Alexnet architecture consists of a dataset of 227*227*3 RGB images as input, convolution layer, pooling layer, active layer, dropout layer and a fully connected layer. And at last a softmax classifier is used to obtain classification results. But Alexnet algorithms are too dependent on large amounts of data and hence require a lot of time for training. So, the authors of [6] proposed use of transfer learning to overcome this problem. The main objective of transfer learning is to transfer information between relevant sources and target classes. The main idea behind using transfer learning is to train the model using parameters from a large dataset and use the model to classify a small amount of test data in order to achieve high accuracy. In the proposed model in [6], Alexnet is used to train the model on Image dataset to obtain the weight parameters through a series of convolution pooling processes and then the model is saved after optimal results are obtained. Then the saved optimal model is shifted to the data with less samples for training. After fine-tuning the model can be used for classification.

Alexnet-tranfer+svm					
result / label	bus	dinosaur	elephant	flower	horse
bus	100%	0.0%	0.0%	0.0%	0.0%
dinosaur	0.0%	85%	10%	0.0%	5.0%
elephant	0.0%	0.0%	90%	0.0%	10%
flower	0.0%	0.0%	0.0%	100%	0.0%
horse	0.0%	0.0%	0.0%	0.0%	100%

Project	AlexNet-tranfer+svm	Hog+SVM	AlexNet
Time	2 m	17.56 s	8 h

Fig. 9: Classification Accuracy and Classification Time

The system proposed in [6], was able to achieve an accuracy of 95%. It also avoided the problem of overfitting caused in deep learning. The model was able to reduce the training time from a few hours to a few minutes. However the model only solves the problem of under-adaptation if provided with a dataset. Hence before using the model in real life applications, a solution is needed to fix the problem of under-adaptation in the dataset and negative transfer in the Alexnet model. The generalization ability i.e. the ability to adapt to the given dataset of the transfer learning model is very poor.

The authors proposed a system in [7], to classify images into two categories either cat or dog. Their work mainly focussed classifying images into respective categories, resolving the overfitting issue and attaining high performance. The dataset used by the authors consists of 10000 images of cats and dogs. It contains 5000 images belonging to each class and they split the dataset into 4:1 proportion for training and testing purposes. They used a three layered convolutional neural network for feature extraction and classification.

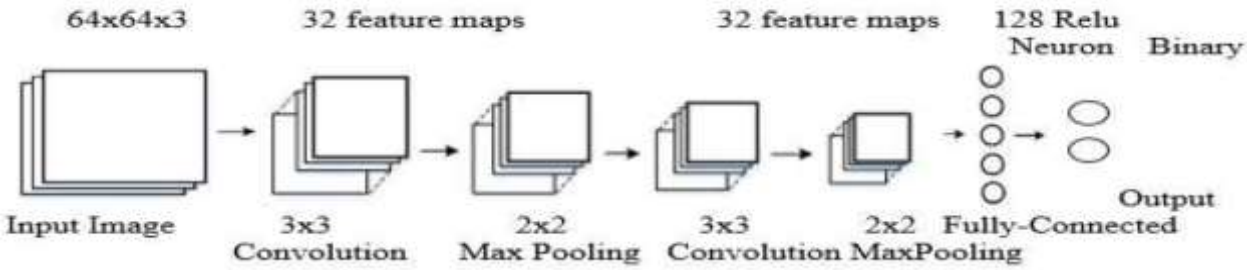

Fig.10: CNN Architecture

The CNN used for feature extraction consists of alternating convolution and pooling layers. The proposed architecture in [7], consists of two convolutional-pooling layers, one fully connected layer along with one input and one output layer. The input contains 64*64*3 neurons, specifying the RGB values of images. Both the convolutional-pooling layers contain kernels of size 3*3 with stride of length 1 pixel to separate out the 32 feature maps, this maps are then passed to a max pooling layer of region 2*2. The next fully connected layer obtains the output from the second conv-pooling layer. It consists of 128 neurons which are connected to each other. The activation function used in this layer and previous layer is ReLU. The output of this layer is then passed to the final layer which has 2 neurons which correspond to the classification category, cat or dog. The CNN model described in [7], is developed with the help of Theano, Keras and TensorFlow which are popular deep learning frameworks.

The authors ran the model on the given training dataset for 25 and 50 epochs. For 25 epochs the number of images which were misclassified was high. Also the testing accuracy achieved was only 66%. But for 50 epochs the model achieved training accuracy of 84.45%. Also the overfitting issue was diminished after 50 epochs. The authors of [7], were able to discover that adding more layers in the CNN architecture and using a much larger dataset can help to gain better testing and training accuracy.

Food classification from images is considered as one of the most challenging tasks in image classification domain. The authors of [8], described a model using deep learning with CNN to classify food-11 dataset. The model of CNN used is V3 inception.

The dataset food-11 contains 16643 images categorized into 11 classes. The classification classes are Bread, Dairy products, Egg, Fried Food, Meat, Pasta, Rice, Seafood, Soup and Vegetables. The dataset is divided into 3 parts: training set, validation set and testing set. The authors applied some preprocessing techniques on the images to increase the processing time and also to fit the images in the V3 inception model. They converted all the input images to 299*299*3 size. Then they used ZCA whitening on all the images. It helps to reduce the redundancy in the matrix of pixel images and highlights the structures and features in the image to CNN. Then they applied random rotation, horizontal or vertical shifts and random flip techniques to the images.

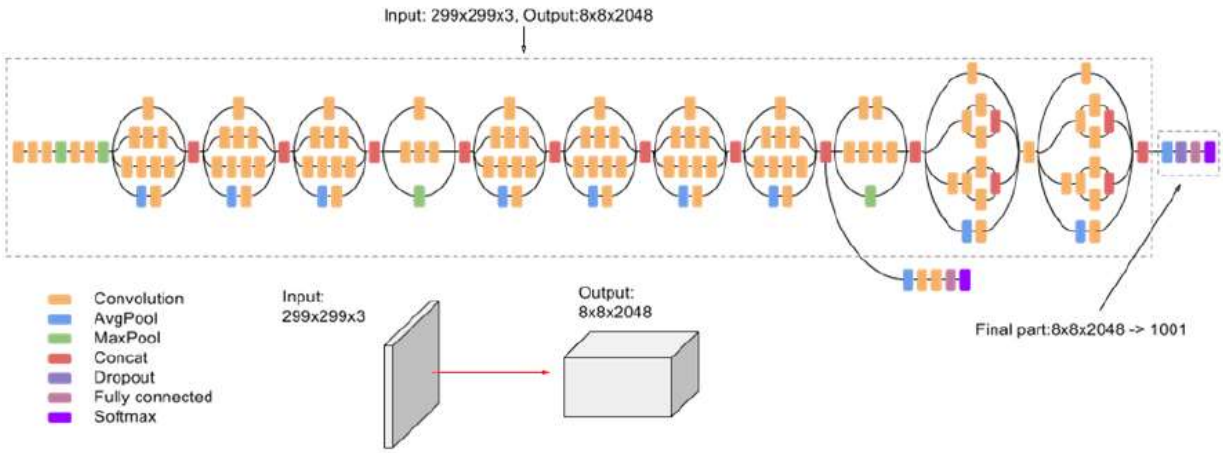

Fig. 11: Inception V3 Architecture

The system in [8], uses an inception V3 model. The model has 7 layers. The convolution layer accepts images of size 299*299*3 and by convoluting these images creates feature maps. The max pooling layer performs the discretization process on the feature maps received from the convolution layer as input. It extracts most important features such as vertical edges, horizontal edges of images, etc. The average pooling layer tries to reduce the variance and complexity in the data obtained from the previous layer. The concat layer concatenates multiple input blobs present to a single output blob. The dropout layer randomly drops the neurons from the layers of CNN. It's main function is to improve performance against overfitting. The output of the dropout layer is then passed to the fully connected layer. The fully connected layer represents the feature vector which is essential for classification. Finally, the

softmax layer assigns probabilities to each class in case of multi-class classification. Depending on the probabilities assigned to each class the one which is closest to 1 is considered as the final class of the input image.

The proposed system in [8], is able to achieve an accuracy of 92.86%, as some already defined algorithms or models such as Alexnet and Caffenet achieved 82.23% and 80.12% accuracy. At initial epochs overfitting can be observed in the model. Even though the system has high performance there's still a considerable gap between training and testing accuracies. The authors also discovered that by using feature based models the computation time can be reduced. Other algorithms such as RNN or DCNN will also give similar performance, was one of the main discoveries made by the authors.

All the above described systems are state of the art methods that have been introduced for solving the problem of image classification using deep learning. They utilize the resources provided to them wisely and manage to produce good results depending on the features available in the images.

V. CHALLENGES AND FUTURE WORK

1. Overfitting in CNN

Overfitting applies to a condition under which a model learns statistical regularities unique to the training set, i.e., instead of learning the signal, it ends up memorizing the unrelated noise, and thereby does less well on a corresponding new dataset. As an overfitted model is not generalizable to never-seen-before results, this is one of the key challenges in machine learning. Compared to the validation set, if the model performs well on the training set, then the model has obviously been overfitted with the training results.

2. Lots and lots of data

Using data, deep learning algorithms are trained to learn gradually. In order to ensure that the machine achieves desirable outcomes, large data sets are required. The analogous artificial neural network takes a huge volume of data, like the human brain needs a lot of experience to understand and deduce knowledge. The more efficient abstraction you want the more parameters need to be tuned and the more data required for more parameters. A neural network's complexity can be expressed by the number of parameters. In the case of deep neural networks, this amount will exceed millions, tens of millions, or even hundreds of millions in certain cases. For example, a speech recognition system will require data from multiple accents, time scales, and demographics.

3. Hyperparameter optimization

The parameters whose value is specified prior to the start of the learning process are hyperparameters. A big improvement in the output of the model can be invoked by adjusting the value of those parameters by a small number. It can have a major effect on model efficiency by depending on the default parameters and not doing Hyperparameter Optimization. Getting too few hyperparameters and hand tuning them instead of maximizing them by validated methods is also a motivating feature of efficiency.

4. High performance hardware

It takes a lot of data to train a data set for a Deep Learning approach. The computer has to be fitted with enough computing capacity to execute a task to address real life problems. Data scientists are transitioning to multi-core high-performance GPUs and related processing units to ensure greater efficiency and less time usage. These units of production are expensive and use a lot of electricity. Thus the application of Deep Learning solutions to the real world is an expensive and energy-consuming affair.

5. Neural Networks are Blackboxes

We know our model parameters, and we feed the neural networks with known data and how they are put together. Yet we don't generally understand how they arrive at a given solution. Neural networks are simply Balckboxes and it is difficult for researchers to grasp how they deduce conclusions.

It is difficult to incorporate high-level cognitive functions because of the lack of neural networks' capacity to reason on an abstract level. Their activity is therefore largely invisible to humans, making them unfit for domains in which process verification is necessary.

6. CNN’s lack multitasking and flexibility

When trained, Deep Learning models can offer an incredibly effective and precise approach to a complex problem. In the present landscape however, the architectures of the neural network are particularly specialized in particular domains of operation.

Many of our programs are working on this theme and are extremely successful at solving one problem. Also addressing a very close topic needs retraining and re-evaluation. Researchers are working hard to build models of Deep Learning that can multitask without the need to rework the whole architecture.

- **Future Scope**

To resolve the problem of overfitting multiple methods have been proposed, but use of more training data significantly reduces overfitting in the model. Because, a model trained on a large dataset achieves the generalization ability efficiently.

Improving the performance of a CNN is a very tedious task. But by minimizing the number of parameters and increasing the size of the dataset it can be achieved.

Most of the systems that are designed today are using CPU as the main processing element, it reduces the cost of the operation but increases the time consumption of the system. Hence use of GPU based systems is also under research, which will save time

and also increase the performance of the model. Also increasing the memory of the CPU will help in reducing the time taken by the system for training.

However using Progressive Neural Networks, there are minor developments in the multi-tasking aspect of neural networks. There is also substantial development in Multi Task Learning. A neural network architecture focused on the performance of vision, language and audio networks, was proposed by researchers to jointly solve a variety of problems covering various realms, including image recognition, translation and speech recognition. This will obviously help in designing the neural networks capable of multi tasking.

VI. CONCLUSION

The use of deep learning and convolutional neural networks is only going to rise in the future. As they provide a way to classify images efficiently. CNN's are also used successfully in other domains such as speech recognition, NLP, text recognition and object recognition.

The previous systems designed for image classification used algorithms such as Bayes theorem, SVM for classification and HSV, HE models for feature extraction which were time consuming and didn't provide better performance. But use of CNN in place of traditional approaches has significantly increased the performance of image classification systems as well as reduced the time requirement.

In this paper, we carried out a survey to visualize how deep learning approaches such CNN, RESNET, Alexnet can be applied to the image processing domain. We reviewed some state of the art systems designed to classify images using above mentioned approaches. We managed to review different types of CNN's such as Alexnet, RESNET, Capsnet, inception V3,etc and observe their performance on different image datasets. And we can definitely conclude that use of deep learning in image processing has been proven to be more efficient than traditional methods. CNN's have definitely improved the classification accuracies of various models that we reviewed in this paper.

Based on our knowledge, we provided insights on different challenging areas of research such as overfitting in neural networks, size of datasets, hyperparameter optimization, requirement of high performance hardware and multi tasking ability of CNN's which should be further explored in the near future.

REFERENCES

[1] Sai Yeshwanth Chaganti, Ipseeta Nanda, Koteswara Rao Pandi, Niraj Kumar,"*Image Classification using SVM and CNN*," in 2020 IEEE Xplore.
[2] Turan Goktug Altundogan, Mehmet Karakose, "*Image Processing and Deep Neural Image Classification Based Physical Feature Determiner for Traffic Stakeholders*," in 2019 IEEE.
[3] Zhiyong Dong ,Sheng Lin, "Research on image classification based on Capsnet," 2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference, IEEE, 2019.
[4] Mrs. Arpana Mahajan, Dr. Sanjay Chaudhary, "*Categorical Image Classification Based On Representational Deep Network (RESNET)*," in 2019 IEEE Third International Conference on Electronics Communication and Aerospace Technology, IEEE, 2019.
[5] Shyava Tripathi, Rishi Kumar, "Image Classification using small Convolutional Neural Network" in 2019, IEEE.
[6] Yunyan Wang, Chongyang Wang, Lengkun Luo, Zhigang Zhou, "Image Classification Based on transfer Learning of Convolutional neural network" in 2019 38th Chinese Control Conference, IEEE, 2019.
[7] Santisudha Panigrahi, Anuja Nanda, Tripti Swarnkar, *"Deep Learning approach for Image Classification,"* in 2018 2nd International Conference on Data Science and Business Analytics, IEEE, 2018.
[8] Md Tohidul Islam, B.M. Nafiz Karim Siddique, Sagidur Rahman, Taskeed Jabid, *"Image Recognition with Deep Learning,"* in 2018 ICIIBMS, IEEE, 2018.
[9] Aditya Vailaya, Mário A. T. Figueiredo, Anil K. Jain, Hong-Jiang Zhang, "*Image Classification for Content-Based Indexing*," in 2001 IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE, 2001
[10] Ardhendu Bandhu, Sanjiban Sekhar Roy, "*Classifying multi-category images using Deep Learning : A Convolutional Neural Network Model*," in 2017 2nd IEEE International Conference On Recent Trends in Electronics Information & Communication Technology, IEEE, 2017.
[11] Deepika Jaswal, Sowmya.V, K.P.Soman, "*Image Classification Using Convolutional Neural Networks*," in 2014, IJSER.
[12] S. Regina Lourdhu Suganthi, Dr. Hanumanthappa, Dr. S. Kavitha, "*Event Image Classification using Deep Learning*," in 2018 International Conference on Soft-computing and Network Security, IEEE, 2018.
[13] M Manoj krishna, M Neelima, M Harshali, M Venu Gopala Rao, "*Image classification using Deep learning*," in 2018, IJET.
[14] Junho Yim, Jeongwoo Ju, Heechul Jung, and Junmo Kim, "*Image Classification Using Convolutional Neural Networks With Multi-stage Feature*," in 2015, Springer International Publishing Switzerland.
[15] Rajiv Jain, Curtis Wigington, "*Multimodal Document Image Classification*," in 2019 International Conference on Document Analysis and Recognition, IEEE, 2019.
[16] Evgin Goceri, "*Analysis of Deep Networks with Residual Blocks and Different Activation Functions: Classification of Skin Diseases*," in 2019, IEEE.