INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 6.078
(Volume 6, Issue 6)
Available online at: https://www.ijariit.com

Natural scene image classification using CNN

Jayanth H. N.
jayanthhn022@gmail.com
New Horizon College of Engineering, Bengaluru, Karnataka

ABSTRACT

Research mainly focused on CNN model for feature extraction and classification of Images. Convolutional Neural Network (CNN) has demonstrated promising performance in image classification tasks. In this project, the algorithm is used to classify the images or natural scenes into 6 classes. This model at last predicts the accuracy or probabilities of different class labels and this probability is used for the predicting class at the end. This dataset is used for both training and testing purpose. It provides the accuracy rate 84.93%. Images with combination of two scenes create and ambiguity hence it is difficult for model to classify. Therefore, it leads to failure in algorithm sometimes. Images used in the training purpose are RGB images. The computational time for processing these images is relatively high as compare to other normal images. Stacking the model with more layers and training the network with more image data using clusters of GPUs provide more accurate results of classification of images.

Keywords: *Image Classification, CNN, RGB Image*

1. INTRODUCTION

1.1 Problem statement

In this problem, a dataset of images from a wide range of natural scenes from all around the world is provided. The task is to identify which kind of scene can the image be categorized into. This model used should be able to do the scene classification task, trained on dataset helping in image processing and recognition in much accurate way as not done earlier.

1.2 General-Introduction

How do we humans, recognize a forest as a forest or a mountain as a mountain? We are very good at categorizing scenes based on the semantic representation and object affinity, but we know very little about the processing and encoding of natural scene categories in the human brain.

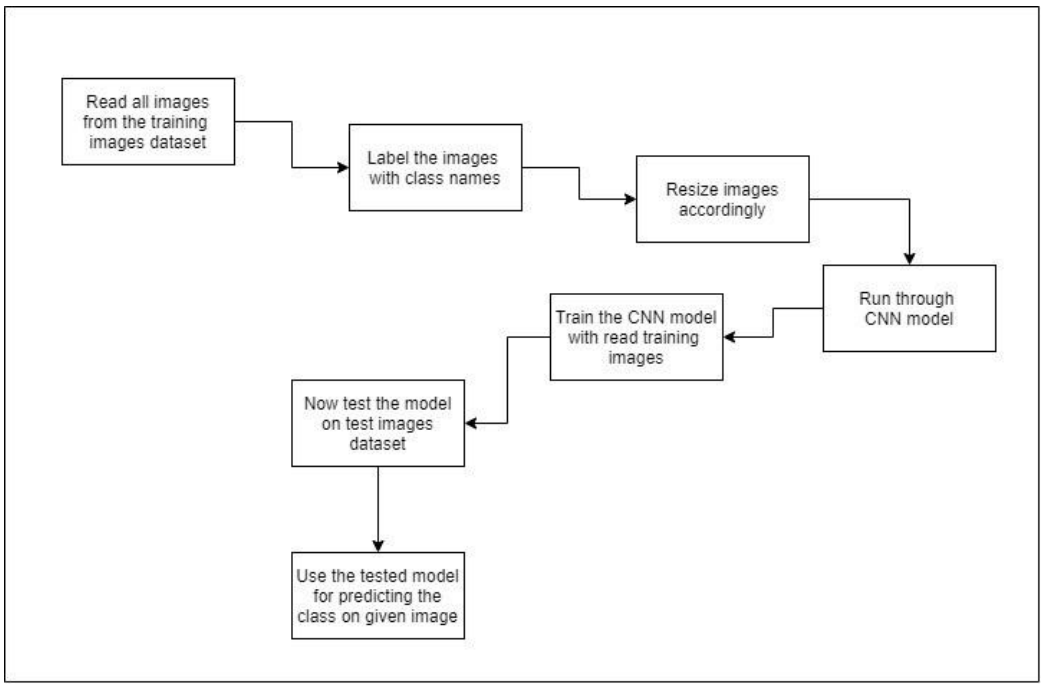
Natural scene perception refers to the process by which an agent (such as a human being) visually takes in and interprets scenes that it typically encounters in natural modes of operation. Classification is the process of categorizing a stimuli into a finite set of classes or labels. The process normally involves recognition of the dominant content in a scene. The dominant content gets the strongest confidence score irrespective of the transformation of that content such as scaling, location or rotation.

In recent years, use of multimedia technology has increased tremendously. In multimedia technology image is one of the important part and image can have different contents in it, such as face, human, scene, text, etc. Realworld scene perception involves sensory and cognitive processing of visual input that in some important sense is like that typically encountered during the natural course of everyday human activity. Unlike objects, which are viewed from a vantage point outside of themselves, scenes are specific views of the environment within which the viewer is embedded. This volume brings together an eclectic group of investigators, all of whom study critical issues in the perception of true real-world scenes.

In the machine learning literature, the term ”natural scene” is usually intended as the one of a semantically coherent, nameable human-scaled view of an outdoor real world environment , and the term ”natural scene categorization” refers to the task of grouping images into semantically meaningful categories . Natural scene categorization is without doubts a difficult task and an open research field, similarly identifying natural scenes from all around the world is an interesting computer vision problem. In this project, model is going to classify six different category Images(buildings, forest, glacier, mountain, sea, street).

To achieve this goal, one of the famous machine learning algorithms out there which is used for Image Classification i.e. Convolutional Neural Network(or CNN). So basically what is CNN – as known it’s a machine learning algorithm for machines to understand the features of the image with foresight and remember the features to guess whether the name of the new image fed to the machine. They have applications in image and video recognition, recommender systems, image classification, medical image analysis, and natural language processing.

1.3 General Block Diagram



1.4 Applications

- Image understanding: When images can be automatically understood and indexed by computer, the efficiency of running digital libraries and video database system will be greatly improved
- Self-Learning: This model can be embedded in any application such as in mobile where, small kids with very much little knowledge about nature can take a picture of scene. This model helps them to recognise the scene.
- Due to the recent advances in satellite sensors, a large amount of highresolution remote sensing images is now being obtained each day.This model can be used to classify those images.
- Visual impaired people can also use it. When they go for sight seeing, this technology can help them in monitoring the scenes as they click the image. The captured image will be classified using this model and with help of voice assistance present in device, can help them to recognise the scene(classified by technology) by speaking up the name of it.

1.5 Challenges

- Processing Time of Image Indexing: Image classification can take hours to process as multiple categories need to be referenced into the output.Solutions must maintain fast processing speeds that are relevant for time-sensitive investigations.
- Scene complexity: The scene provided for processing may be affected with the weather conditions such as fog,which might cover the dominant feature in the image required for classification.
- If the given natural scene image is degraded due to noise or might be blurred which makes it difficult to identify and classify.
- There might be cases where multiple scene can be in a single image creating problem or confusion in model to classify. Taking an example of if the image with both street and building is processed,there will be ambiguity in the model for classification.

1.6 Motivation

- Identifying the natural scenes through the computer rather than human vision is interesting thing to work on,as we see these scenes in day to day lives.
- Classification of images,in the field of computer vision is itself a challenging task.CNN is the best technique for doing the same giving the best accuracy.
- Nature is best gift and most beautiful thing most of the people would love to see and experience making it much more interesting topic to think about

1.7 Objectives

- The main objective is to classify the event/scene in the image using the classification model as well as provide a semantic labels to the scene environment within the image.
- Studying the basic principles of Image Recognition, and understanding the practical applications with state of art facilities and tremendous future possibilities.

2. LITERATURESURVEY

Natural scene classification is an open problem in computer vision and has wide applications in both image and video indexing.

- Maron et al, in 1998 used Multiple-Instance learning to classify images of natural scenes, which is highly relevant to my topic. This learning method is a way of modelling ambiguity in supervised learning examples.

- Rong Yan, Yan Liu, Rong Jin and A.Hauptmann, in 2003 publised paper on predicting rare classes with SVM ensembles in scene classification.In this paper, they proposed SVM ensembles to address the rare class problem. Various classifier combination strategies are investigated, including majority voting, sum rule, neural network gater and hierarchical SVMs.This experimental results show that hierarchical SVMs can achieve significantly better and more stable performance than other strategies, as well as high computational efficiency.
- Matthew et al,in 2004 presented a framework to handle the classification error problem and apply it to the problem of semantic scene classification. Their paper was based on the fact that a natural scene may contain multiple objects such that the scene can be divided into multiple categories. Their method was demonstrated on the SVM classifier.

3. PROPOSED METHOD

3.1 Design

Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 148, 148, 200)	5600
conv2d_1 (Conv2D)	(None, 146, 146, 180)	324180
max_pooling2d (MaxPooling2D)	(None, 29, 29, 180)	0
conv2d_2 (Conv2D)	(None, 27, 27, 180)	291780
conv2d_3 (Conv2D)	(None, 25, 25, 140)	226940
conv2d_4 (Conv2D)	(None, 23, 23, 100)	126100
conv2d_5 (Conv2D)	(None, 21, 21, 50)	45050
max_pooling2d_1 (MaxPooling2	(None, 4, 4, 50)	0
flatten (Flatten)	(None, 800)	0
dense (Dense)	(None, 180)	144180
dense_1 (Dense)	(None, 100)	18100
dense_2 (Dense)	(None, 50)	5050
dropout (Dropout)	(None, 50)	0
dense_3 (Dense)	(None, 6)	306

Total params: 1,187,286
Trainable params: 1,187,286
Non-trainable params: 0

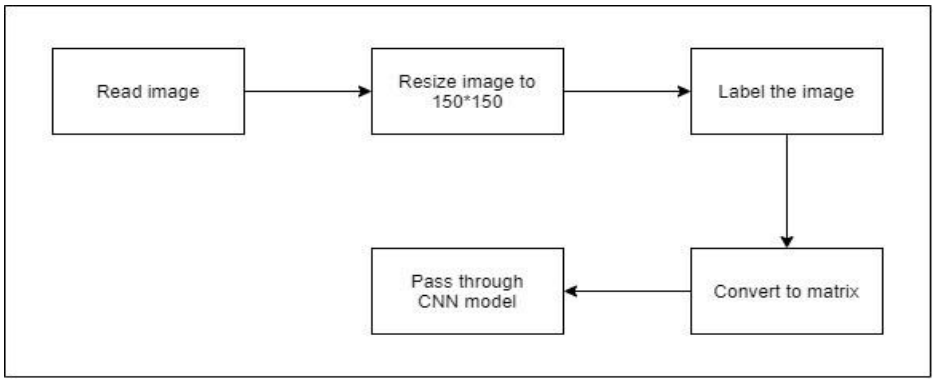
- Convolution- Kernel size of (3,3) with stride 0
- Max pooling - Pool size of (5,5) with stride 5
- Dense(Fully connected layer) - Using softmax activation function

My CNN model contains 15 layers. This is because the images are made up of complex features. In the architecture aspect, it is composed of convolutional layer, maxpooling layer and output layer. Firstly, a convolutional layer with 200 channels is used to expand the number of feature maps to show the feature better. Then the number of channels continuously decreases as 200→ 180→ 180→ 140 → 100→ 50→ 50 to reduce the calculation complexity. For the pooling part, two pooling layers are involved to reduce the spatial size as third and sixth layers.

Four fully connected layers are introduced to decrease the length of image vector to 6 which represents 6 classes. The dimension of vector decreases slowly as 800→ 180→ 100→ 50→ 6. The reason is if the dimension of vector quickly drops from 800 to 6, they could be information loss.

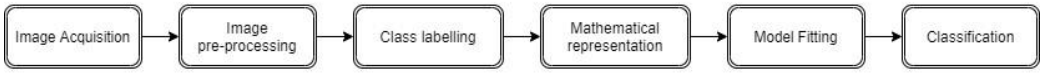
Activation function for the last fully connected layer is softmax and ReLU for the remaining layers. ReLU function works for images as the inputs are non-negative values and also improves the sparsity. Softmax function gives output a list of probabilities for corresponding classes.

Block diagram



3.2 Algorithm

We use CNN as it convolves learned features with input data and uses 2D convolutional layers. This means that this type of network is ideal for processing 2D images. Compared to other image classification algorithms, CNNs actually use very little preprocessing as shown is section 3.1 block diagram. This below diagram shows the **sequence of algorithm**



Basic Algorithm for proposed model

Step 1: **Image Acquisition:** It is the process of capturing an image from any object using a digital camera.This image is stored in RGB format in dataset. The RGB image is read from the dataset.

Step 2: **Image pre-possessing:** Some image features are pre-processed to enhance important features. It is used for additional processing and study. Resize the image to a definite size and noise is removed.This is one of the important step.

Step 3: **Class Labelling:** Label the image using one hot encoder. The pre-processed image is labelled under six classes present in the dataset for further processing.

Step 4: **Mathematical representation:** Using NumPy images are converted to array and then it is converted to matrix format with numeric values. Image analysis techniques are comprised of Co-occurrence Matrix and calculations of associated texture feature.

Step 5: **Model Fitting:** Fit the matrix into series of convolution layers and max pooling. Matrix is used to examine the textures that considers the spatial relationship between pixels in co-occurrence matrix. All important featured are extracted by model while training with dataset.

Step 6: **Classification:** Trained CNN model is given the set of images for testing. Accuracy of the model is obtained in this phase by validating the model with prediction dataset. Results are concluded with labelling the images along with displaying the probabilities of each class respectively for each image. After deriving results, stop the process.

Step 7: **Post-processing:** Extracting patterns,draw trends and deriving conclusion for further usage

3.3 Required Mathematical Equation

$$\sum_{x=0} f(x-\tau)h(\tau \qquad (1)$$

where, f(x) is a discrete function representing image. h(x) is a filter.

$$image(x,y) \otimes filter(x,y) = {}^{X}({}^{X} f(x-a,y-b)h(x,y))) \qquad (2)$$

$$y=0\ x=0$$

This operator mimics the "sliding" of our filter over every region where it fits on the image, and returns the feature map.

$$outputmap = (((inputimage \otimes F_1) \otimes F_2) \otimes F_3) \qquad (3)$$

If the input image is suppose to be convolve with 33 filter. Then, convolve the output feature map with another 3x3 filter, and do this one more time to get a final feature map. We will denote the filters as F and feature maps as M.

$$outputmap = inputimage \otimes (F_1 \otimes F_2) \otimes F_3) \qquad (4)$$

The output of the input image and filter F1 will me feature map M1, so we can simply substitute to get this statement only in terms of filters and the input image. The critical idea here is that the convolution operator is associative.

$$y = \max(0,x)$$

(RELU activation function)

The function we apply to the feature map outputted by each convolution is called an activation function. Convolutional layers almost exclusively use the activation function known as RELU.

4. EXPERIMENTAL ANALYSIS

4.1 Information on data set

The dataset consists of the natural scene images with 24,343 as total number of images. This dataset is mainly divided into 6 categories-

1) Building
2) Forest
3) Glacier
4) Mountain
5) Sea
6) Street.

Dataset is sub divided into three parts, test dataset, train dataset and prediction dataset. Each dataset is composed of images.
Test dataset and train dataset are further divided into 6 categories above mentioned. Each class has appropriate same number of images. Train dataset consists of total 14042 images and test dataset with total 3000 images. The images are RGB pictures with size 150*150.

The prediction dataset has 7301 number of images. These images are not categorized into categories because it is used to predict the class labels on the images present in it. It also has RGB pictures with size 150*150.

4.2 Experimental Settings

At first this model is executed using one of the online compiler i.e. colab and the programming language is python. The experimental framework is Tenserflow and Sklearn.

The using the natural image scene dataset tests this model effectiveness. This dataset consists of the images that are categorized into 6 classes/categories (building, forest, glacier, mountain, sea and street). These six categories are differentiated by substituting each of them with an unique number such as, building→ 0, forest→ 1, glacier→ 2, mountain→ 3, sea→ 4 and street→ 5. These are class labels assigned to each of the category in ease of computation. This process took place following the algorithm as mention in 4.2.

To train this model,train dataset as mention in 5.1 was used. In this 70% of the train dataset was used for training and 30% for partial testing with batch size of 32 for both process. Next this model was tested with large samples around 3000 samples.

After training and testing, prediction dataset as mentioned in 5.1 was used for predicting the class label or classifying the images present in this dataset.

The results show that, each image, with the label on top and predicted probablities at bottom for each class was displayed. Result table(5.3) show the experimental result of the model with classification accuracy of it.

4.3 Result Table

	precision	recall	f1-score	support
0	0.81	0.84	0.83	437
1	0.94	0.97	0.96	474
2	0.83	0.78	0.70	553
3	0.79	0.82	0.80	525
4	0.87	0.86	0.87	510
5	0.86	0.84	0.85	501
accuracy			0.85	3000
macro avg	0.85	0.85	0.85	3000
weighted avg	0.85	0.85	0.85	3000

- This section shows the recorded result as shown above.
- Classification report is obtained after performing all the steps. This report is tabulated in a table.
- This result table show the shows the precision value, recall value, f1score value and support value for each class.
- Confusion matrix is also built to infer results.

4.4 Discussion on Result
Confusion Matrix

	Building	Forest	Glacier	Mountain	Sea	Street
Building	369	5	1	4	8	50
Forest	1	462	3	4	1	3
Glacier	3	6	432	83	25	4
Mountain	5	6	61	429	21	3
Sea	13	5	23	24	437	8
Street	66	6	1	1	8	419

To elaborate the results, let us consider the confusion matrix as shown above. The labels on the y-axis are TRUE LABELS and on x-axis are PREDICTED LABELS. This matrix basically gives the idea of how many true predictions have been made with respect to the true labels.

The 1st row of the of the confusion matrix shows that, the model predicted scene as building 369 times when the actual image was building, 5 times when the actual scene was forest, 1 time when actual scene was glacier, 4 times when it was mountain, 8 times when it was sea and 50 times when it was street image. From these values we can draw conclusion or result that the scene is building because it has maximum value with respect to others.

In similar manner from next rows we can draw result that, 462 times forest was predicted when the scene was actually forest. Diagonal values represent the number of predictions with the true label of same class (same scene).

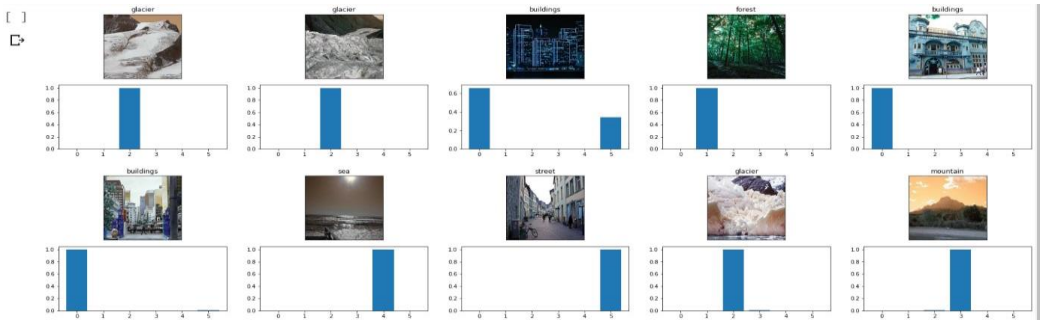
From result table we can infer that the proposed model gave accuracy of 84.93%, which is almost equal to 85% on this dataset with 15 layers of CNN. f1-score is the harmonic mean of recall and precision. This result is obtained by considering the fact that images in dataset have noise in them. Results show that this algorithm failed to classify glacier (class label 2) as it has least recall value in the table.

4.5 Algorithm complexity
To find the complexity of this algorithm is a difficult task since they are mutli-dependent factors to consider. The factors may include large size of image, the amount of noise present in them or the number of factors we try to extract. It is quite difficult to obtain the complexity considering these complexities n the algorithm.

In this model the complexity basically depends on the number of images used to train the model. If it is large number, like in my dataset it is around 14000 images to be trained so time constraint can not be taken under consideration. Since we are using CNN, the epoch layers also plays a role in the time complexity. In order to attain higher accuracy,many layers are created in the model. But this again takes a lot of time to execute. Hence, we can conclude that the dependent factors to find the time complexity of algorithm are-

1) Dataset size.
2) Layers in the model.
3) Batch size while training model
4) No. of epochs.

4.6 Snapshots of Project Result



Accuracy of Model

```
[ ] test_images,test_labels = get_images('/content/drive/My Drive/intel/seg_test/seg_test/')
    test1_images = np.array([i for i in test_images]).reshape(-1,150,150,3)
    test2_labels = np.array([i for i in test_labels])
    model.evaluate(test1_images,test2_labels, verbose=1)
```

```
Done with buildings
Done with forest
Done with sea
Done with street
Done with glacier
Done with mountain
94/94 [==============================] - 4s 45ms/step - loss: 0.7033 - accuracy: 0.8493
[0.7032608389854431, 0.8493333458900452]
```

Confusion Matrix

```
[ ] from sklearn.metrics import confusion_matrix,classification_report
    cm=confusion_matrix(test2_labels,predicted)
    cm
```

```
array([[369,   5,   1,   4,   8,  50],
       [  1, 462,   3,   4,   1,   3],
       [  3,   6, 432,  83,  25,   4],
       [  5,   6,  61, 429,  21,   3],
       [ 13,   5,  23,  24, 437,   8],
       [ 66,   6,   1,   1,   8, 419]])
```

Classification Report

```
[ ] print(classification_report(test2_labels,predicted))
```

	precision	recall	f1-score	support
0	0.81	0.84	0.83	437
1	0.94	0.97	0.96	474
2	0.83	0.78	0.80	553
3	0.79	0.82	0.80	525
4	0.87	0.86	0.87	510
5	0.86	0.84	0.85	501
accuracy			0.85	3000
macro avg	0.85	0.85	0.85	3000
weighted avg	0.85	0.85	0.85	3000

5. CONCLUSION & FUTURE SCOPE

This report summarizes the applications and challenges of this project. Manual extraction of feature is very difficult and is time consuming, therefore CNN is the best to replace the manual feature extraction as it is highly versatile in feature extraction.

In this project, I used Convolutional Neural Networks (CNN) for classification of images present in Natural scene image classification dataset. This dataset is used for both training and testing purpose. It provides the accuracy rate 84.93%. Images with combination of two scenes creates and ambiguity hence it is difficult for model to classify. Therefore, it leads to failure in algorithm sometimes. Images used in the training purpose are RGB images. The computational time for processing these images is relatively high as compare to other normal images. Stacking the model with more layers and training the network with more image data using clusters of GPUs will provide more accurate results of classification of images.

The future enhancement will focus on classifying the colored images of large size and also add some more different category scenes like waterfall, sky, etc and work on it. Also try to recognize and classify the scenes in video.

6. REFERENCES

[1] Maron, Oded, and Aparna Lakshmi Ratan. ”Multiple-Instance Learning for Natural Scene Classification.” ICML. Vol. 98. 1998.

[2] Rong Yan, Yan Liu, Rong Jin and A.Hauptmann.”On predicting rareclasses with SVM ensembles in scene classification”.2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP ’03).pp-1520-6149

[3] Boutell, Matthew R., et al. ”Learning multi-label scene classification.”Pattern recognition 37.9 (2004): 1757-1771.

[4] Bosch,Anna,AndrewZisserman,andXavierMuoz.”Sceneclassification viapLSA.” European conference on computer vision. Springer, Berlin, Heidelberg, 2006.
[5] : Grossberg, S., Huang, T.R. (2009). ARTSCENE: A neural system for natural scene classification. Journal of Vision, 9(4):6, 1–19,
[6] Deepika Gupta, Ajay Kumar Singh, Deepa Kumari, Raina. ”NatureScene classification using different color feature”.Vol. 2, Issue 3, MayJun 2012, pp.937-941
[7] Jason Holloway, Tanu Priya, Ashok Veeraraghavan and Saurabh Prasad.”Image classification in natural scenes: Are a few selective spectral channels sufficient?” (2014) IEEE International Conference on Image Processing (ICIP)
[8] Dixit, Mandar, et al. ”Scene classification with semantic fisher vectors.” Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.
[9] Hu, Fan; Xia, Gui-Song; Hu, Jingwen; Zhang, Liangpei. 2015. ”Transferring Deep Convolutional Neural Networks for the Scene Classification of High-Resolution Remote Sensing Imagery.” Remote Sens. 7, no. 11: 14680-14707.
[10] Cheng, Gong, Junwei Han, and Xiaoqiang Lu. ”Remote sensing imagescene classification: Benchmark and state of the art.” Proceedings of the IEEE 105.10 (2017): 1865-1883
[11] Aparna R. Rout and Sahebrao B. Bagal.”Natural Scene Image Classification Using Deep Learning”. 2017 International Conference on Computing, Communication, Control and Automation (ICCUBEA).
[12] C.A.Laulkar and P.J.Kulkarni.”Outdoor Natural Scene Object Classification Using Probabilistic Neu-ral Network”. International Journal of Computer Sciences and Engineering- Research Paper,Volume-6, Special Issue-1, Feb 2018 E-ISSN: 2347-2693
[13] Himanshu Patel and Hiren Mewada.”Analysis of Machine LearningBased Scene Classification Algorithms and Quantitative Evaluation”.
[14] International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 10 (2018) pp. 7811-7819.

