INTERNATIONAL JOURNAL OF ADVANCED INNOVATIVE TECHNOLOGY IN ENGINEERING

Published by Global Advanced Research Publication House Journal

Home page: www.ijaite.co.in

CNN Algorithm Visualizer

[1]Abishai Shankpal, [2]Azhan Khan, [3]Sahil Sarkar, [4]Saquib Sheikh, [5]Shoaib Shaikh, [6]Sushant Menon, [7]Prof. Syed Rehan

[1,2,3,4,5,6,7]Department Computer Science & Engineering, Anjuman College of Engineering and Technology, RTMNU Nagpur, Maharashtra, India

[1]abishaishankpal14@gmail.com, [2]khanazhan9@gmail.com, [3]sahilsarkar2306@gmail.com, [4]saquibsheikh094@gmail.com, [5]shoaibshaikh1102@gmail.com, [6]sushantnambiar@gmail.com, [7]srehan@anjumanengg.edu.in

Article History

Received on: 8 April 2022

Revised on: 18 April 2022

Accepted on: 09 May 2022

Keywords: Energy Consumption, Supercapacitor, Metro Train System, Transportation System

e-ISSN: 2455-6491

Production and hosted by
www.garph.org

ABSTRACT

Image classification is widely used in various fields such as classification of diseases on leaves, facial expressions classification. To make large images realistic, classify images implemented using the concept of deep neural networks. Suggestion The work implemented the InceptionV3 model to classify an image into a category such as living thing are further classified into layers like animals, people. The article brings methodology for more accurate image classification image feature extraction or image segmentation.

1. INTRODUCTION

Deep neural networks are a widely used technique in areas such as self-driving, healthcare, autonomous machines Translations, etc. Classification of images using Neural networks is a recent approach to obtaining good results. The network can be used to classify the image dataset into different classes using pre-trained InceptionV3 models. Extracting features from images is computationally expensive, the pre-trained neural network is the best choice for image classification. We used InceptionV3 as the pre-format model that classifies images into four classes, each of which has about 2000 images. The image dataset is preprocessed for the first time then the InceptionV3 model is trained. Pictures may vary in size, so they are preprocessed before they can be used for the training model. Specifically, convolutional neuron networks trained against large volumes of annotated data are capable of top results. We will analyze the results of the experiments with InceptionV3 and also explore some of the different performances of pre-trained models.

A. Convolution Neural Network Layers

A convolutional neural network consists of the following layer.

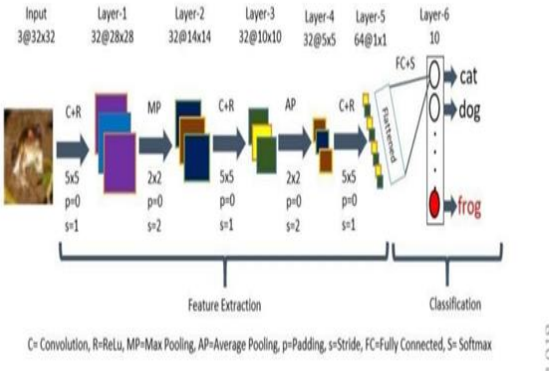


Figure 1: Layers of CNN

Convolution Layer: A convolution layer is the main building block of a CNN. It contains a set of filters (or kernels) whose parameters will be learned during training. The filter size is usually smaller than the actual image. Each filter corresponds to the image and generates an activation map. For convolution, the filter is slid over the height and width of the image, and the dot product between each filter element and the input is calculated at each position. The output volume of the convolution layer is generated by superimposing the activation maps of each filter along the depth dimension.

Pooling Layer: To recognize the pattern present in an image, let's suppose an image of a cat, the network must recognize it as a cat, whether it is walking, jumping, standing still, or running. Image flexibility is required, and that's where the pooling layer comes in. The maximum pooling is in which we swipe through the feature map and extract tiles of the specified size. For each cell, the largest value is output to a new feature map and all other values are ignored. It works with image measurements to gradually reduce the size of the input image so that objects in the image can be detected and identified wherever they are. Pooling also helps control "overfitting". Maximum pooling is reducing the size of the image to get more information.

Fully connected layer: Fully connected layer in CNN represents function input vectors. This feature vector or tensor or class has important information for entry. When the network trains, this feature vector is then used for classification, regression, or making an entry to another network like Recurrent Neural The network translates to another output type and so on. He is also used as an encoded vector. During training, this function vector is used to calculate the loss and help the network for him to be trained. Transformation classes above all the connected layers maintain information about local features in the input image i.e., edges, blobs, shapes, etc. This class maintains multiple symbolic filters for one of the local features. The fully connected layer is still composite and composite information from all important convolutional layers.

2. Related Work

Legitimate Neural Networks (CNNs) are used in several tasks with excellent performance in various applications. Handwritten digit recognition was one of the first applications where the CNN architecture was successfully implemented. Since the inception of CNN, there has been a continuous improvement in networks with the innovation of new layers and the involvement of different computer vision techniques. Convolutional neural networks are mainly used in challenging image networks with various combinations of sketch datasets. Few researchers have shown a comparison between human subjects and the detection ability of a network trained on image datasets. The comparison results showed that humans matched an accuracy rate of 73.1% on the dataset, while results from a trained network showed an accuracy rate of 64%. Similarly, when Convolutional Neural Networks were applied to the same dataset it yielded an accuracy of 74.9%, hence outperforming the accuracy rate of humans. The used methods mostly make use of the strokes' order to attain a much better accuracy rate. There are studies going on that aim at understanding Deep Neural Network's behavior in diverse situations. These studies present how small changes made to an image can severely change the results of grouping. The work also presents images that are fully unrecognized by human beings but are classified with high accuracy rates by the trained networks. There have been many developments in the field of feature detection and description and many algorithms and techniques have been developed for object and scene classification. Overall, we draw attention to the similarity between object detectors, texture filters, and filter banks. There is a lot of work in the literature on object detection and scene classification. The researchers mainly used the currently updated descriptor from Felzenszwalb and the context classifier from Hoeim. The idea of developing different object detectors to interpret images is essentially similar to the work done in the multimedia community, in that they use a large number of "semantic concepts" for image and video annotations and semantic indexing. In the literature related to our work, each semantic concept is formed using an image or video frame.

A. Drawbacks Of Existing Systems

While we found that CNN Visualizer provided participants with an engaging and enjoyable learning experience and made it easier for them to get used to CNN, we also saw several potential improvements to the program. with our current system design from this study. Beginners need more guidance. We found that participants with little knowledge of CNN's needed more instruction to get started using CNN Visualizer. Some participants reported that CNN's representation of images and animations was not easy to understand at first, but the instructional article and text annotation helped them greatly in interpreting the visualizations. Limited explanation of why CNN works. Some participants, especially those with little experience with CNNs, are interested in knowing why the CNN architecture works in addition to learning how the CNN model makes predictions.

3. Proposed Methodology

Deep Learning has become a major tool for self-perception problems such as understanding images, human voices, and robots exploring the world. We aim to implement the concept of a convolutional neural network for image recognition. Understanding CNN and applying it to the image recognition system is the goal of the proposed model. Neural Convolutions network extracts feature maps from 2D images using filters. The accumulative neural network considers the mapping of image pixels to the neighboring space rather than having a fully connected layer of neurons. The Convolutional neural network has been proved to be a very dominant and potential tool in image processing. Even in the fields of computer vision such as handwriting recognition, natural object classification, and segmentation, CNN has become a much better tool compared to all other previously implemented tools. When starting to learn deep learning with neural networks, it was realized that one of the most supervised deep learning techniques is convolutional neural networks. We design a complex neural network to realize visual patterns directly from pixel images with minimal pre-processing. Almost all CNN architectures follow the same general design principle of successively applying complexity classes to the input, periodically downsampling (maximum clustering) the spatial dimensions when increasing the number of object maps. In addition, there are also fully connected layers, activation functions, and loss functions (e.g., cross-entropy or softmax). However, among all CNN operations, convolutional layers, group layers, and fully connected layers are the most important. Therefore, we will briefly introduce these classes before presenting our proposed model. The convolution layer is the first layer where it can extract features from the image. Because pixels are related only to pixels that are adjacent and close to each other, convolution allows us to maintain relationships between different parts of an image. Convolution is filtering the image with a smaller pixel filter to decrease the size of the image without losing the relationship between pixels. When we apply convolution to a 7x7 image by using a filter of size 3x3 with a 1x1 stride (1pixel shift at each step), we will end up having a 5x5 output. When constructing CNN, it is common to insert pooling layers after each convolution layer, so that we can reduce the spatial size of the representation. This layer reduces the parameter counts, and thus reduces the computational complexity. Also, pooling layers help with the overfitting problem. We choose the cluster size to reduce the number of parameters by choosing the mean and maximum clustering operation. A fully connected network is in any architecture where each parameter is linked to one another to determine the relation and effect of each parameter on the labels. We can vastly reduce the time-space complexity by using the convolution and pooling layers. We can construct a fully connected network, in the end, to classify our image's sum values inside these pixels. A simple convolutional network is a sequence of layers. The layer transforms one volume of activations into another through a differentiable function. We use three main types of layers to build our network architecture. It is a composite layer, a group layer, and a fully connected layer. We will stack these layers to form six layers of network architecture.

A. Dataset Description

In this research, we have used two datasets that are (Cat vs Dog) and (Female vs Male).
(Cat vs Dog) the dataset comprises 25000 images and (Female vs male) comprises 12500 images.

Table 1: Sample dataset

Dataset	Images	Train Images	Test Images
Cat-vs-Dog	25000	12500	12500
Female-vs-Male	12500	5700	6800

B. Pre-Processing

At first, we need some pre-processing on the images like resizing images, normalizing the pixel values, etc. After the necessary pre-processing, data is ready to be fed into the model. Layer-1 consists of the convolutional layer with the relu (Rectified Linear Unit) activation function which is the first convolutional layer of our CNN architecture. This layer gets the pre-processed image as the input of size n*n=32*32. The convolutional filter size (f*f) is 5*5, padding (p) is 0(around all the sides of the image), stride (s) is 1, and the number of filters is 32. After this convolution operation, we get feature maps of size 32@28*28 where 32 is the number of feature maps that is equal to the number of filters used, and 28 comes from the formula ((n+2p-f)/s) +1= ((32+2*0- 5)/1) +1=28. Then the relu activation is done in each feature map. Layer-2 is the max-pooling layer. This layer gets the input of size 32@28*28 from the previous layer. The pooling size is 2*2; padding is 0 and stride is 2. After this max-pooling operation, we get feature maps of size 32@14*14. Max pooling is done in each feature map independently, so we get the same number of feature maps as the previous layer and 14 comes from the same formula ((n + 2pf)/s) +1. This class has no activation function. Layer3 is the second convolutional layer with a relu activation function. This layer takes 32@14*14 size input from the previous layer. The filter size is 5*5; the buffer is 0, the stride is 1, and the filter count is 32. After this convolution operation, we get feature maps of size 32@10*10. Then relu activation is done in each feature map. Layer4 is the average pooling layer. This layer gets the input of size 32@10*10 from the previous layer. The pooling size is 2*2; padding is 0 and stride is 2. After this max-pooling operation, we get a feature map of size 32@5*5. Layer5 is the third convolutional layer with a relu activation function. This layer gets the input of size 32@5*5 from the previous layer. The filter size is 4*4; padding is 0, the stride is 1, and the number of filters is 64. After this convolution operation, we get feature maps of size 64@1*1. This layer acts as a fully connected layer and produces a one-dimensional vector of size 64 when flattened. Layer6 is the last layer of the network. It is a fully connected layer. This layer will compute the class scores, resulting in a vector of size 10, where each of the ten numbers corresponds to a class score, such as among the ten categories of the CIFAR10 dataset. For final outputs, we use the softmax activation function. In this way, CNN transforms the original image layer by layer from the main pixel values to the final class scores. Note that some classes contain parameters and some do not. In particular, the full convolution/connection layers perform transformations that are a function not only of the activations in the input volume but also of the parameters (neuron weights and biases). On the other hand, the Relu/aggregate classes will perform the freeze function. We train the parameters in fully connected/integrated layers with random gradient origin. Through this process, we will prepare the trained model that will be used to recognize the images contained in the test data. Therefore, we can categorize images into the cat, dog, and human gender.

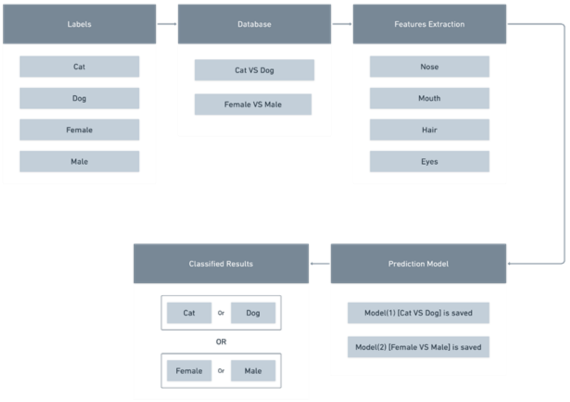
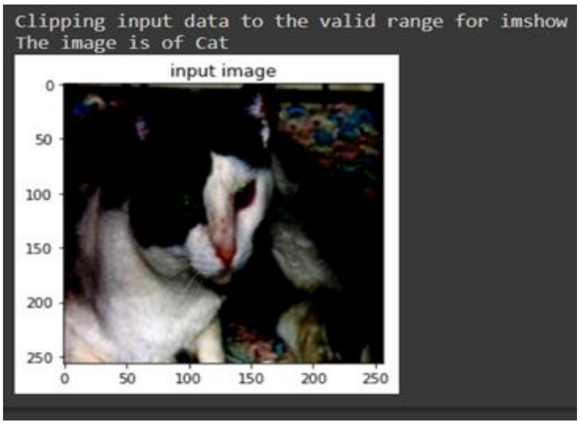


Figure 2: Flowchart of Image Classification

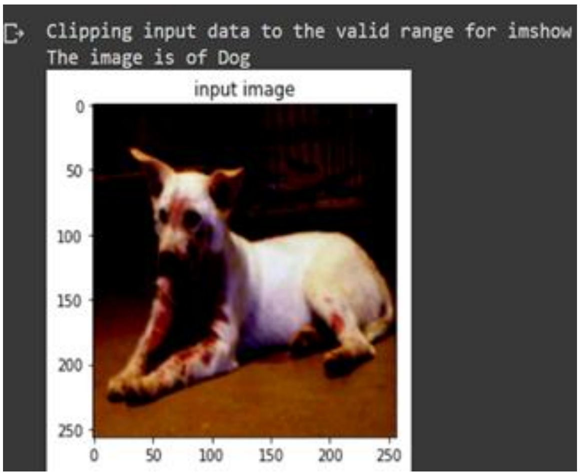
4. Result Analysis

This section presents the results of the classification accuracy obtained using the CNN algorithm on different standard datasets. Results are presented using percent classification accuracy for the in-train and test data separately. With the percentage value of classification accuracy. The idea here is that using a sufficient number of epochs will result in low MSE, high classification accuracy, and minimal time to train the network. The network is tested on different data sets, each of which is tested for a different number of iterations (epochs) in turn.



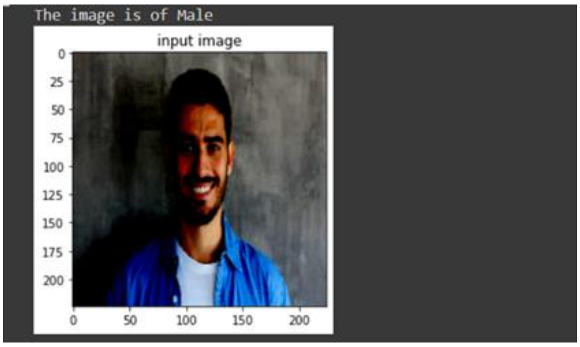
```
Clipping input data to the valid range for imshow
The image is of Cat
```

Figure 3: Classification of Cat



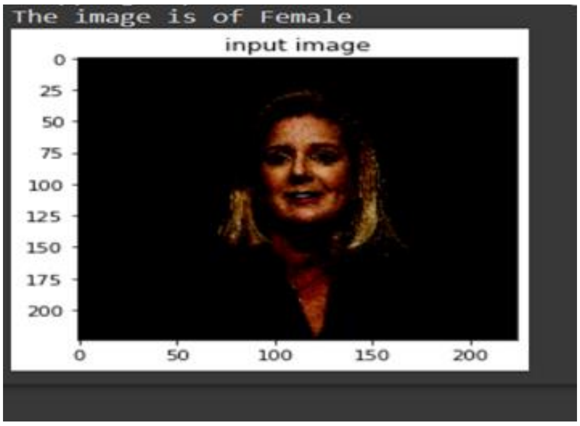
```
Clipping input data to the valid range for imshow
The image is of Dog
```

Figure 4: Classification of Dog



```
The image is of Male
```

Figure 5: Classification of Male



```
The image is of Female
```

Figure 6: Classification of Female

This study used two datasets i.e. (Cat vs Dog) and (Female vs Male) and got an accuracy of 95% and 87% respectively.

Table 2: Accuracy Score

Images	Accuracy
Cat vs Dog	95%
Female vs Male	87%

5. Conclusion

The first major conclusion from this project is that deep learning is an extremely powerful tool. The representative power of neural networks is amazing, and they have been specialized for use in computer vision and natural language processing, turning tasks that appear very hard, or even impossible, into reasonable ones. We have seen that in CNNs, the depth of the model has a strong correlation with performance. Tuning hyperparameters also proved to boost the performance of CNN models by a good amount.

6. Future scope

It might be possible to carry out further research in this area only using GPU servers, unlike personal computing devices.

Conflict Of Interest

The authors declare that they have no conflict of interest.

Funding Support

The author declares that they have no funding support for this study.

References

[1] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov," Scalable Object Detection Using Deep Neural Networks," 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, 2014, pp. 2155-2162.

[2] Day O, Khoshgoftaar T M . A survey on heterogeneous transfer learning[J]. Journal of Big Data, 2017, 4(1):29.

[3] Wang Wenpeng, Mao Wentao, He Jianliang, et al. Smoke Recognition Method Based on Deep Migration Learning [J]. Computer Applications, 2017 (11): 144-149+161(in Chinese).

[4] Planas, Santiago, et al. "Performance of an ultrasonic ranging sensor in apple tree canopies." Sensors11.3 (2011): 2459-2477

[5] LI Yandong, HAO Zongbo, LEI Hang. Survey of the convolutional neural network[J]. Journal of Computer Applications, 2016, 36(9): 2508-2515.

[6] Nature-http://arti.vub.ac.be/research/colour/data/imagesets.zip

[7] Animal-Group https://www.kaggle.com/search?q=ANIMALS