

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

Volume 3, Issue 2, June 2023

# **Deep CNN Image Classifier**

Prof. Manoj Chittawar<sup>1</sup>, Tisha Sorte<sup>2</sup>, Ujma Mohammad<sup>3</sup>, Alveena Sheikh<sup>4</sup>, Ragini Kamble<sup>5</sup>, Saundarya Meshram<sup>6</sup>

Professor, Computer Science Engineering Department<sup>1</sup> Students, Computer Science Engineering Department<sup>23456</sup> Rajiv Gandhi College of Engineering, Research and Technology, Chandrapur, Maharashtra, India

**Abstract:** Many problems in computer vision were saturating on their accuracy before a decade. However, with the rise of deep learning techniques, the accuracy of these problems drastically improved. One of the major problem was that of image classification, which is defined as predicting the class of the image. Cat and Dog image classification is one such example of where the images of cat and dog are classified. This paper aims to incorporate state-of-art technique for object detection with the goal of achieving high accuracy. A convolutional neural network is been build for the image classification task.

Keywords: Image Classification, Convolutional Neural Network, Keras, Deep Learning, Callbacks

# I. INTRODUCTION

Lillsand and Kiefer defined image processing as involving manipulation of digital images with the use of computer. It is a broad subject and involves processes that are mathematically complex. Image processing involves some basic operations namely image restoration/rectification, image enhancement, image classification, images fusion etc. Image classification forms an important part of image processing. The objective of image classification is the automatic allocation of image to thematic classes. Two types of classification are supervised classification and unsupervised classification. The process of image classification involves two steps, training of the system followed by testing. The training process means, to take the characteristic properties of the images (form a class) and form a unique description for a particular class. The process is done for all classes depending on the type of classification problem; binary classification or multi-class classification. The testing step means to categorize the test images under various classes for which system was trained. This assigning of class is done based on the partitioning between classes based on the training features. Since 2006, deep structured learning, or more commonly called deep learning or hierarchical learning, has emerged as a new area of machine learning research. Several definitions are available for Deep Learning; coating one of the many definitions from Deep Learning is defined as: A class of machine learning techniques that exploit many layers of nonlinearinformation processing for supervised or unsupervised feature extraction and transformation and for pattern analysis and classification. This work aims at the application of Convolutional Neural Network or CNN for image classification.

The Dogs vs. Cats image classification has been around for a long time now. The Dogs vs. Cats competition from Kaggle is trying to solve the CAPTCHA challenge, which relies on the problem of distinguishing images of dogs and cats. It is easy for humans, but evidence suggests that cats and dogs are particularly difficult to tell apart automatically.

Many people has worked or are working on constructing machine learning classifiers to address this problem. A classifier based on color features got 56.9% accuracy on the Asirra dataset. An accuracy of 82.7% was achieved from a SVM classifier based on a combination of color and texture features.

In our project we are going to build a convolutional neural network to solve the problem and achieve higher performance and better results. In our project instead of using the Kaggle data set comprising of total 25000 images, I would be working on subset of these images. My dataset would be comprising of total 10000 images. Keras would used for model building and all the code would be implemented on google colab.

# **II. CONVOLUTIONAL NEURAL NETWORK**

The name "convolutional neural network" indicates that the network employs a mathematical (convolution) operation. Convolution is a specialized kind of linear operation. Convolutional networks are simply neural networks that use

Copyright to IJARSCT www.ijarsct.co.in DOI: 10.48175/IJARSCT-11313





International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

#### Volume 3, Issue 2, June 2023

convolution in place of general matrix multiplication in at least one of their layers. A convolutional neural network consists of an input and an output layer, as well as multiple hidden layers. The hidden layers of a CNN typically consist of a series of convolutional layers that convolve with a multiplication or other dot product. The activation function is commonly a RELU layer, and is subsequently followed by additional convolutions such as pooling layers, fully connected layers and normalization layers, referred to as hidden layers because their inputs and outputs are masked by the activation function and final convolution. The final convolution, in turn, often involves backpropagation in order to more accurately weight the end product.



Figure 1. Basic CNN Architecture

# **III. WORKING OF THE IMAGE CLASSIFICATION**

#### 3.1 Dataset

The first step is to download the dataset of dog vs cat from Kaggle. After downloading the dataset from Kaggle, one zip file will be downloaded. Then we will unzip the zip file and will extract the content from the zip file. In this zip file, there are two folders, one is test and one is train. There are 4000 images of each cat and dog for training and 1000 image of each cat and dog for testing. The images are of varing shape and sizes, but in order to train a CNN the images should be of same size.

# 3.2 Libraries

After extracting the images, some libraries we need to import. Two important libraries are used i.e.tensorflow and keras. These libraries helps us to deal with image classification. After importing libraries, the images are divided into training data and validation data.

#### **3.3 Model Architecture**

Model is the name given to the Sequential model. The model's first layer is a Conv2D layer. Since, it is the first layer of the model, input shape of the images that are going to be supplied to the model is being mentioned. Next layer is abatch normalization layer. Then one activation layer corresponding to the conv2d layer. Further there is another set of conv2d, batch normalization and activation layer with different number of kernels in the conv2d layer. After that a max Pooling layer is there and then a dropout layer is there.

The same set of layers is again repeated with different number of kernel's and dropout rate. The convolution layers end with this set. Next are the fully connected layer.

The Sequential model API is used to build model. The sequential API allows you to create models layer-by-layer. The 'add()' function to add layers to our model. The model needs to know what input shape it should expect. For this reason, only the first layer in a Sequential model needs to receive information about its input shape.

Dropout layer consists in randomly setting a fraction rate of input units to 0 at each update during training time, which helps prevent overfitting.

Activation layer are used to apply the activation function to the output of that layer. The purpose of the activation function is to introduce non-linearity into the output of a neuron. Relu and sigmoid activation are used in the model.

Batch Normalization is used for improving the speed, performance, and stability of artificial neural networks. Batch normalization is a method we can use to normalize the inputs of each layer and achieve faster convergence.

Copyright to IJARSCT www.ijarsct.co.in DOI: 10.48175/IJARSCT-11313





International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

#### Volume 3, Issue 2, June 2023

Maximum pooling, or max pooling, is a pooling operation that calculates the maximum, or largest, value in each patch of each feature map. The results are down sampled or pooled feature maps that highlight the most present feature in the patch.

In Conv2D layer, a kernel, convolution matrix, or mask is a small matrix. It is used for blurring, sharpening, embossing, edge detection, and more. This is accomplished by doing a convolution between a kernel and an image. This layer creates a convolution kernel that is convolved with the layer input to produce a tensor of outputs.

Then comes the fully connected layers. It contains only 2 layers. First one is the global average pooling layer to minimize overfitting by reducing the total number of parameters in the model. Second layer and the final layer is the Dense layer with sigmoid activation.

Dense layer implements the operation:  $output = activation(dot(input + kernel) + bias, activation is the element-wise activation function passed as the activation argument, kernel is a weights matrix created by the layer, and bias is a bias vector created by the layer (only applicable if use_bias is True).$ 

There are total 14,848,193 total parameters, out of which 14,847,745 are trainable parameters and 448 are non-trainable parameters. The major number parameters are form the conv2d layer. Batch normalization and dense layer also contribute few of the parameters.

### 3.4 Model Compilation and Training

During the model compilation, the optimizer algorithm, loss function and the list of metrics are parameters which are to be taken care of. Adam is used as the optimization algorithm, binary cross entropy is used as the loss function and accuracy is the only metric used. Early Stopping and ModelCheckPointercallbacks are used to prevent overfitting and save the best state of the model. These callbacks are mentioned as a list during the training. Sequential models fit\_generator() is used to train the model. Model is trained for 10 epochs with EarlyStopping and modelCheckPointercallbacks.

+ Text 📣 Copy to Drive						
· · · · · · · · · · · · · · · · · · ·	×=	· · · ·				
history = model.fit(train_ds,epochs=18	,validation_data=	validation_c	ls)			
Epoch 1/10						
525/625 [====================================	=] - 71s 112ms/ste	ep - loss: 1	.2870 - accuracy:	0.6077 - val_loss:	0.6443 - val_accuracy:	0.538
Epoch 2/10						
525/625 [==============================	=] - 70s 112ms/ste	ep - loss: @	3.5226 - accuracy:	0.7400 - val_loss:	0.5353 - val_accuracy:	0.744
Epoch 3/10						
25/625 [====================================	=] - 71s 113ms/ste	ep - loss: @	.4516 - accuracy:	0.7917 - val loss:	0.4704 - val accuracy:	0.78
poch 4/10						
525/625 [====================================	=] - 70s 111ms/ste	ep - loss: @	.3935 - accuracy:	0.8224 - val loss:	0.7385 - val accuracy:	0.562
Epoch 5/10						
525/625 [====================================	=] - 70s 111ms/ste	ep - loss: @	.3376 - accuracy:	0.8543 - val loss:	0.5054 - val accuracy:	0.770
Epoch 6/10	-					
525/625 [====================================	=] - 70s 112ms/ste	ep - loss: @	.2517 - accuracy:	0.8946 - val loss:	0.8225 - val accuracy:	0.742
Epoch 7/10						
525/625 [====================================	=] - 70s 112ms/ste	ep - loss: @	.1735 - accuracy:	0.9294 - val_loss:	0.5472 - val_accuracy:	0.798
poch 8/10						
25/625 [====================================	=] - 70s 111ms/ste	ep - loss: @	.1280 - accuracy:	0.9551 - val loss:	0.6331 - val accuracy:	0.804
poch 9/10						
25/625 [	=] - 70s 111ms/ste	ep - loss: 0	.0917 - accuracy:	0.9671 - val_loss:	0.6862 - val_accuracy:	0.801
poch 10/10	o X					
525/625 [====================================	=1 - 70s 111ms/ste	en - loss: @	0.0755 - accuracy:	0.9744 - val loss:	0.7166 - val accuracy:	0.80

#### Table 1 : Training Result Per Epoch

plt.plot(history.history['accuracy'],color='red',label='train')
plt.plot(history.history['val\_accuracy'],color='blue',label='validation')

import matplotlib.pyplot as plt

# **3.5 Model Evaluation**

The model trained for 15 epochs after which it stopped due to the presence of EarlyStoppingcallback which had the patience parameter set to 5. The training accuracy kept on increasing but the validation accuracy started to decrease which might be due to overfitting. That was the reason EarlyStopper check pointer was used to prevent results obtained due to overfitting. Below is the table showing the end result of training i.e train accuracy, test accuracy, train loss, test loss and epochs.

The graph below shows training accuracy and testing accuracy vs the number of epochs.

Copyright to IJARSCT

DOI: 10.48175/IJARSCT-11313



ISSN 2581-9429 IJARSCT



International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

#### Volume 3, Issue 2, June 2023

### Chart-1: Accuracy vs Epoch

It is evident from the graph that the train accuracy kept on increasing with the number of epochs but the test accuracy become near constant after the 11th epoch.

The graph below shows the train loss and test loss vs the number of epochs.



#### Chart-2: Loss vs Epoch

It is evident from the graph that the train loss kept decreasing with the number of epochs but the test loss become near constant after 11th epoch.

The final accuracy was obtained by initializing the model with the weights that were stored during training by the use of ModelCheckPointercallbacks. Final accuracy of 90.10% was obtained on the testing data.



# **IV. CONCLUSION**

In this paper we build a deep convolutional neural network for image classification (cat and dog images). Despite of using only a subset of the images an accuracy of 90.10% was obtained. If the whole dataset was being used the accuracy would have been even better.

# V. ACKNOWLEDGEMENT

We would like to express our deepest gratitude and respect to Z. J. Khan, Principle, RCERT and Prof. Dr. Nitin Janwe, Head of Department of Computer Science and Engineering, RCERT, Chandrapur, Maharashtra. We are grateful to Prof. Manoj Chittawar our Project Guide for their precious assistance and guidance. We received enormous support and help from our professors. Thanking them for their valuable feedbacks. We could not have undertaken the journey of this project without all guidance and experience they shared with us.

Copyright to IJARSCT www.ijarsct.co.in DOI: 10.48175/IJARSCT-11313





International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

#### Volume 3, Issue 2, June 2023

### REFERENCES

[1] Golle, P. (2008, October). Machine learning attacks against the Asirra CAPTCHA. In Proceedings of the 15th ACM conference on Computer and communications security (pp. 535-542). ACM.

[2] J. Elson, J. Douceur, J. Howell and J. Saul. Asirra: a CAPTCHA that exploits interest-aligned manual image categorization. Proc. of ACM CCS 2007, pp. 366-374.

[3] MuthukrishnanRamprasath, M.Vijay Anand and Shanmugasundaram Hariharan, Image Classification using Convolutional Neural Networks. International Journal of Pure and Applied Mathematics, Volume 119 No. 17 2018, 1307-1319

[4] Parkhi, O. M., Vedaldi, A., Zisserman, A., & Jawahar, C. V. (2012, June). Cats and dogs. In Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on (pp. 3498-3505). IEEE.

[5] Zeiler, M. D., & Fergus, R. (2013). Visualizing and Understanding Convolutional Neural Networks. arXiv preprint arXiv:1311.2901.

[6] Bang Liu, Yan Liu, Kai Zhou "Image Classification for Dogs and Cats".

