Journal of Advanced Research in Technology and Management Sciences (JARTMS)
Volume: 03 Issue: 05 ISSN: 2582-3078 August 2021
Available online at: http:// www.jartms.org

IMAGE CLASSIFICATION USING TENSORFLOW

[1] M. Sri Venkat Rami Reddy, [2]A.Gowthami, [3] Y. D. Solmon Raju
[1] [2] , Assistant Professor, Holy Mary Institute of Technology and Science,
Bogaram, Keesara, Hyderabad, Telangana, India.
[3] Associate Professor & HoD of ECE, Holy Mary Institute of Technology and Science,
Bogaram, Keesara, Hyderabad, Telangana, India.
[1]srivenkatramireddy.m@hmgi.ac.in [2]gowthami.a@hmgi.ac.in [3]davidsolomonraju.y@hmgi.ac.in

Abstract— We developed and trained machine learning model using two popular Deep Learning Libraries which are TensorFlow and KERAS. These two popular libraries are mostly used in real-time because of their flexibility and lucidity. Here we will be using TensorFlow. For this, we need to use Python to work with Tensorflow.

Key words—Tensorflow, Keras, Convolutional Neural Network, Activation functions, Loss functions, Optimizers, Supervised learning.

Journal of Advanced Research in Technology and Management Sciences (JARTMS)
Volume: 03 Issue: 05 ISSN: 2582-3078 August 2021
Available online at: http:// www.jartms.org

I. INTRODUCTION

Image classification has been on the rise and is becoming a trend among technology developers, particularly with data growth in different sectors of industry such as e-commerce, auto, healthcare, and gambling. The most obvious example of this technology is applied to Google Photos. Google Photos can now detect up to 82% accuracy to identify your face with just a few marked and classified pictures in your photo album. The technology itself almost surpasses the human ability to classify or recognize an image. **Keras** is an open-source software library that provides a Python interface for artificial neural networks. Keras acts as an interface for the TensorFlow library.

Machine Vision has a different context when it comes to image classification. The capability of this technology is to recognize people, objects, locations, action and writing in images. The combination of artificial intelligence software and machine vision technologies allows for the exceptional result of image classification.

Deeper neural networks are more difficult to train. We present a residual learning framework to facilitate the formation of networks which are far deeper than those used before. We explicitly rephrase layers as residual learning functions by referring to layer inputs, rather than learning non-state functions.

II. TENSORFLOW

TensorFlow makes it easy for beginners and experts to create machine learning models for desktop, mobile, web, and cloud. See the sections below to get started.

TensorFlow is Google Brain's second-generation system. Version 1.0.0 was released on February 11, 2017. TensorFlow's market share among research papers was declining to the advantage of PyTorch, the TensorFlow Team announced a release of a new major version of the library in September 2019. TensorFlow 2.0 introduced many changes, the most significant being TensorFlow eager, which changed the automatic differentiation scheme from the static computational graph, to the "Define-by-Run" scheme originally made popular by Chainer and later PyTorch. Other major changes included removal of old libraries, cross-compatibility between trained models on different versions of TensorFlow, and significant improvements to the performance on GPU.

III. KERAS

Keras is an open-source software library that provides a Python interface for artificial neural networks. Keras acts as an interface for the TensorFlow library.

Up until version 2.3, Keras supported multiple backends, including TensorFlow, Microsoft Cognitive Toolkit, Theano, and PlaidML.[21][22][23] As of version 2.4, only TensorFlow is supported. Designed to enable fast experimentation with deep neural networks, it focuses on being user-friendly, modular, and extensible. It was developed as part of the research effort of project ONEIROS (Open-ended Neuro-Electronic Intelligent Robot Operating System),[5] and its primary author and maintainer is François Chollet, a Google engineer. Chollet is also the author of the XCeption deep neural network model.

IV. RELATED WORKS

In this project, we will use Convolutional Neural Network for classifying images, since Convolutional Neural Network (CNN) won the image classification competition 202 (ILSVRC12), a lot of attention has been paid to deep layer CNN study. CNN's success is attributed to its superior multi-scale high-level image depictions rather than its low-level manual engineering features.

According to, the journal discussed on image classification system based on a structure of a Convolutional Neural Network (CNN). The training was performed such that a balanced number of face images and non-face images were
used for training by deriving additional face images from the face images data. The image classification system employs
the bi-scale CNN with 120 trained data and the auto-stage training achieves 81.6% detection rate with only six false positives on Face Detection Data Set and Benchmark (FDDB), where the current state of the art achieves about 80% detection rate with 50 false positives.

From the research used Decision Tree (DT) as the techniques in image classification. The DT has multiple datasets that are located under each of Hierarchical classifier. It must be done in order to calculate membership for each of the classes. The classifier allowed some rejection of the class on the intermediary stages. This method also required three parts as the first is to find terminal nodes and the second in placing the class in it. The third is the segmentation of the nodes. This approach is considered to be very simple and effective.

The experiment results prove that the proposed solution will be the greatest tool for dealing with practical problems which are related to use deep CNNs on a small dataset. Our approach not only considerably reduces the need for important training data, but also effectively expands the training data set. The most commonly used parametric classifier is the maximum likelihood classifier (MLC). Contrary to parametric classification, non-parametric classification is not based on statistical hypotheses or parameters.

On the basis of the review, he suggested a rapid classification of images by stimulating fuzzy classifiers. It was a simple way to differentiate between known and unknown category. This method is simply to boost Meta knowledge where local features can be found mainly. It has been tested with a few big image data and compared with the bag-of-features image template. The result gave much better classification accuracy as it was a testing process that gave a short period of time where it produced 30% shorter compared to the previous one.

Name/Year	Title of project	Purpose	Method Used
Gregor, Danihelka, Graves, Rezende, & Wierstra (2015)	DRAW: A Recurrent Neural Network for Image Generation	➢ Train neural network for image classification ➢ Trained complex images with MNIST models	Artificial Neural Network (ANN)
Rastegari, Ordonez, Redmon, & Farhadi (2016)	XNOR-Net: ImageNet Classification Using Binary Convolutional Neural Networks	➢ Balanced number of face images and non-face images are used for training ➢ Employing the bi-scale CNN 120 trained with the auto-stage training	Convolutional Neural Network
Kamavisdar, Saluja, & Agrawal (2013)	A Survey On Image Classification Application Techniques	➢ Multiple dataset that being located under each of Hierarchical classifier ➢ Rejection of the class on the intermediary stage	Decision Tree
Pasolli, Melgani, Tuia, Pacifici, & Emery (2014)	SVM Active Learning Approach for Image Classification Using Spatial Information	➢ Combining spatial information from sequential process of trial process with spectral	Support Vector Machine (SVM)
Korytkowski, Rutkowski, & Scherer (2016)	Fast Image Classication by Boosting Fuzzy Classifiers	➢ Simply boosting Meta knowledge where local characteristic can be mostly found	Fuzzy Classifiers

Table 1. Related works of classification systems.

V. METHOD

Based on figure-1, It is the image classification framework in which deep neural networks are also applied. The process is divided into four (4) phases, each of which will be discussed. Every phase is included on TensorFlow as an open source software and Python as a programming language. Then, the process goes on to collect some of the images (inputs), apply CNN and finally all the images will be classified in their groups.

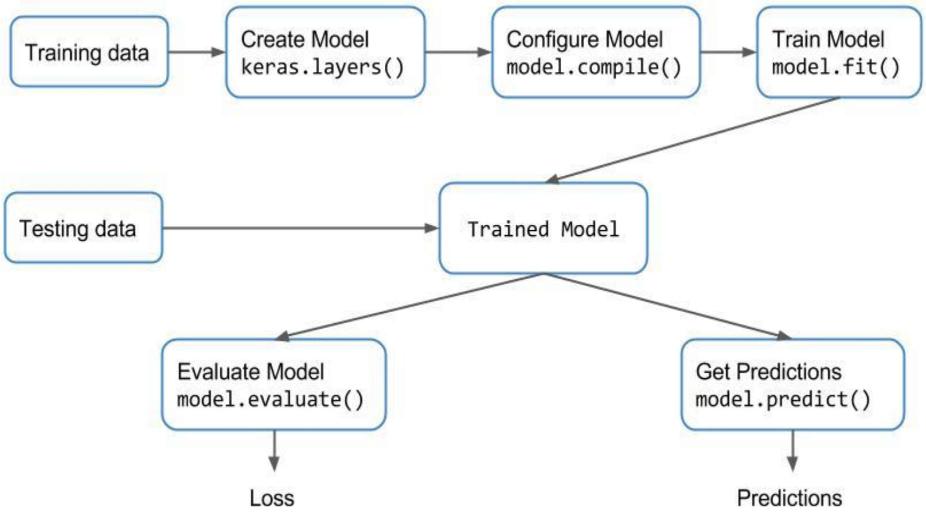


Figure 1. Block diagram for image classification systems

Journal of Advanced Research in Technology and Management Sciences (JARTMS)
Volume: 03 Issue: 05 ISSN: 2582-3078 August 2021
Available online at: http:// www.jartms.org

VI. IMAGE TRAINING

In this paper, training set is what it sounds like. It's the set of data used to train the model. During each epoch, our model will be trained over and over again on this same data in our training set, and it will continue to learn about the features of this data. One of the major reasons we need a validation set is to ensure that our model is not overfitting to the data in the training set. During training, if we're also validating the model on the validation set and see that the results it's giving for the validation data are just as good as the results it's giving for the training data, then we can be more confident that our model is not overfitting.

The input data from this article mostly use thousands of images. All these pictures are extracted from Image Net. ImageNet was also known as the Large Scale Visual Recognition Challenge where it is a competition about detection and classified thousands of objects in its categories. This is a yearly competition from 2010 to the present.

Figure 2. Images of horse from dataset.

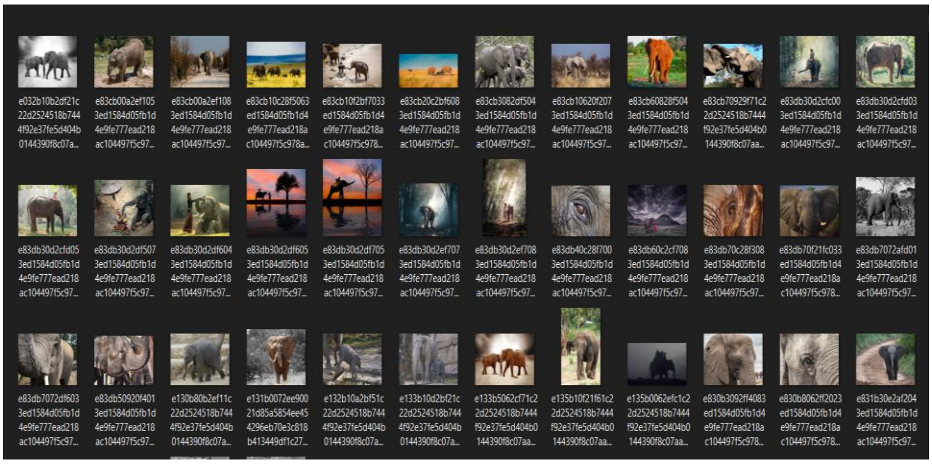


Figure 3. Images of elephant from dataset.

VII. IMPLEMENTATION OF CONVOLUTIONAL NEURAL NETWORK

As shown in Figure, it consists of five (8) data inputs (eight type of different animals) and undergoes training with multiple hidden layers. The inputs are also set with fixed-size of the 224x224 RGB image. The convolution process is configured with MobileNet as it produces efficient convolution neural networks.

Table 2. Number of images to the type of animals

No.	Type of Animal	No of Images
1.	Butterfly	2112
2.	Chicken	3098
3.	Cow	1866
4.	Elephant	1446
5.	Horse	2623
6.	Sheep	1820
7.	Spider	4821
8.	Squirrel	1862
	TOTAL	19658

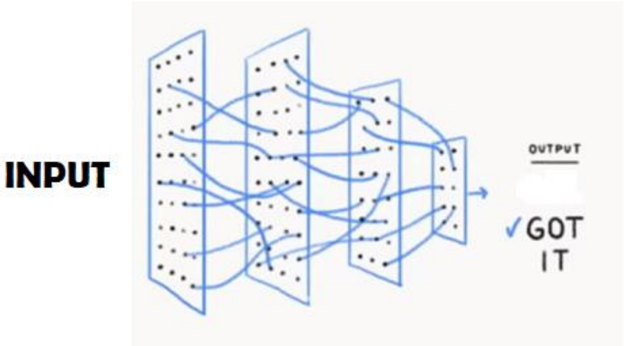


Figure 4. CNN process towards animal images.

VIII. CLASSIFICATION OF SYSTEM

Image classification twill be implemented using TensorFlow.
Python language is used in the programming language. System starts collecting images of animals from the data set. Then ,CNN is applied to train the model. Running for validation or testing and if it is not the image of a particular flower that is supposed to act as output then it should start over from CNN. The process ends when the output is categorized in the right kind of animal. It has five (8) types of the animals which are Butterfly, Chicken, Cow, Elephant, Horse, Sheep, Spider, Squirrel. After that, all of these input images undergo 'training' with the convolutional neural network (CNN).

The convolutional neural network (CNN) had to train all of these sets of data until the systems recognize each of these 19658 images. Then, each of the classifications occurred when one of the images being tested whether it belongs to any of these eight (8) types of animals.

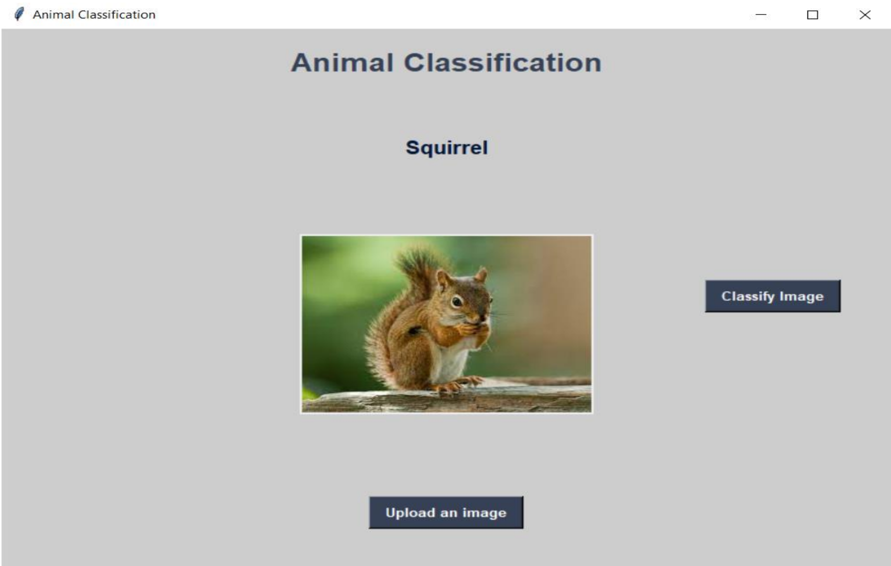
IX. RESULT

When an image is given as input, the model which was trained through Neural Networks detects the feature from the input image and then it predicts the output of the image. the model trained on neural networks in this project is animal classifier which predicts the output with an accuracy of 80%.

A sheep picture is uploaded and then make computer to recognize/categorize the output. The result is categorized according to the model that formed and have a better understanding of the exact appearance of sheep. Then processing takes place and finally the output is predicted.

Journal of Advanced Research in Technology and Management Sciences (JARTMS)
Volume: 03 Issue: 05 ISSN: 2582-3078 August 2021
Available online at: http:// www.jartms.org

A squirrel picture is uploaded and then make computer to recognize/categorize the output. The result is categorized according to the model that formed and have a better understanding of the exact appearance of squirrel. Then processing takes place and finally the output is predicted.



X. CONCLUSION

To conclude, this research focuses on the classification of images using deep learning through the TensorFlow framework. Three (3) objectives were achieved as part of this research. The objectives are linked directly with conclusions because it can determine whether all objectives are successfully achieved or not. . It can be concluded that all results that have been obtained, showed quite impressive outcomes. The convolutional neural network (CNN) becomes the main agenda for this research, especially in image classification technology. CNN technique was studied in more details starting from assembling, training model and to classify images into categories.

TensorFlow also gave good results as it is able to simulate, train and classified with up to 90% of accuracy towards 8-different types of animals. TensorFlow framework which leads to designing of the system involved Python from start until ends.

X. REFERENCES

[1] Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. IEEE Transactions on Neural Networks, (2):157–166, 1994.

[2] C. M. Bishop. Neural networks for pattern recognition. Oxford university press, 1995.

[3] W. L. Briggs, S. F. McCormick, et al. A Multigrid Tutorial. Siam, 2000.

[4] K. Chatfield, V. Lempitsky, A. Vedaldi, and A. Zisserman. The devil is in the details: an evaluation of recent feature encoding methods. In BMVC, 2011.

[5] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The Pascal Visual Object Classes (VOC) Challenge. IJCV, pages 303–338, 2010.

[6] R. Girshick. Fast R-CNN. In ICCV, 2015.

[7] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In CVPR, 2014.

[8] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In AISTATS, 2010.

[9] I. J. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio. Maxout networks. arXiv:1302.4389, 2013.

[10] Dean, Jeff; Monga, Rajat; et al. (November 9, 2015). "TensorFlow: Large-scale machine learning on heterogeneous systems" (PDF). TensorFlow.org. Google Research. Retrieved November 10, 2015.

[11] Oremus, Will (November 9, 2015). "What Is TensorFlow, and Why Is Google So Excited About It?". Slate. Retrieved November 11, 2015.

[12] Ward-Bailey, Jeff (November 25, 2015). "Google chairman: We're making 'real progress' on artificial intelligence". CSMonitor. Retrieved November 25, 2015.

[13] TensorFlow (April 26, 2018). "Introducing Swift For TensorFlow". Medium. Retrieved August 14, 2019. not just a TensorFlow API wrapper written in Swift

[14] "tensorflow: TensorFlow for R". RStudio. February 17, 2018. Retrieved February 18, 2018.

[15] "Tensorflow Release 1.0.0".

[16] "Build and train machine learning models on our new Google Cloud TPUs". Google. May 17, 2017. Retrieved May 18, 2017.

[17] "Cloud TPU". Google Cloud. Retrieved May 24, 2019.

[18] "A MATLAB wrapper for TensorFlow Core". November 3, 2019.
Retrieved February 13, 2020.

[19] He, Horace (October 10, 2019). "The State of Machine Learning Frameworks in 2019". The Gradient. Retrieved May 22, 2020.

[20] "TensorFlow in other languages | TensorFlow Core". TensorFlow. Retrieved August 14, 2019.

[21] "Release 2.4.0". 17 June 2020. Retrieved 18 June 2020.

[22] "Kerasbackends". keras.io. Retrieved 2018-02-23.

[23] Jump up to: "Why use Keras?". keras.io. Retrieved 2020-03-22.

[24] *Vicente, Sara; Rother, Carsten; Kolmogorov, Vladimir (2011). Object cosegmentation. IEEE. doi:10.1109/cvpr.2011.5995530. ISBN 978-1-4577-0394-2.*

[25] Linda G. Shapiro and George C. Stockman (2001): “Computer Vision”, pp 279–325, New Jersey, Prentice-Hall, ISBN 0-13-030796-3

[26] Perez, Sarah (November 9, 2015). "Google Open-Sources The Machine Learning Tech Behind Google Photos Search, Smart Reply And More". TechCrunch. Retrieved November 11, 2015.

XI. ABOUT AUTHORS

Mr. M. Sri Venkat Rami Reddy, Working as Assistant Professor of E. C. E. Dept. in Holy Mary institute of Technology & Science, Hyderabad. He is Graduated B. Tech. in Electronics & Communications Engineering from J. N. T. University, Hyderabad, in the year 2004. He Post-Graduated M. Tech. In V. L. S. I. System Design from J. N. T. University, Hyderabad, in the year 2012. He is CISCO (Routing, Switching & Security) Certified Networking Engineer. His Research interests are V. L. S. I. System Design, SoC, Image Processing, Cyber Security and **A**rtificial **I**ntelligence & **M**achine **L**earning (**AI& ML**). He is a **L**ife **M**ember of **I**ndian **S**ociety for **T**echnical Education (**LM-ISTE**), New Delhi, India.

Ms. A.Gowthami, Working as Assistant Professor of E. C. E. Dept. in Holy Mary institute of Technology & Science, Hyderabad. She Graduated B. Tech. in Electronics & Communications Engineering from J. N. T. Univesity, Kakinada, in the year 2014. She Post-Graduated M. Tech. In DECS from J. N. T. Univesity, Kakinada, in the year 2018. Her Research interests are Antennas, V.L. S. I. System Design, Image Processing, Cyber Security.

Mr. David Solomon Raju is pursuing PhD from Rayalaseema University, Kurnool, Andhra Pradesh in Image Processing. Completed M. Tech in ECE with specialization Systems and Signal Processing from JNTU Hyderabad, in 2007 and B. Tech in Electronics and Communication Engineering from V. R. Siddhartha Engg. College, Nagarjuna University, Guntur, Andhra Pradesh in 2000. Currently working as an Associate Professor and Head of the Dept. in the ECE Department at Holy Mary Institute of Technology and Science (College of Engineering), Hyderabad.

Areas of research interest include Image Processing, Segmentation techniques, Object-Based Compression, Image Sequence Restoration and Enhancement, Cryptography, and Signal Processing. He is a Life Time member of ISTE, ISOI, Fellow Member of IETE, Member of IE and Member of IAENG.