International Journal of Technical Innovation in Modern Engineering & Science (IJTIMES)
Impact Factor: 5.22 (SJIF-2017), e-ISSN: 2455-2585
Volume 5, Issue 07, July-2019

A Deep Machine Learning Neural Network for Real Time Object Classification using Keras & Tensorflow

[1]Vaishali Urkude, [2]Pankaj Pandey

[1]*Computer Science & Engineering, Oriental Institute of Science & Technology, Bhopal, Madhya Pradesh, India*
[2]*Computer Science & Engineering, Oriental Institute of Science & Technology, Bhopal, Madhya Pradesh, India*

Abstract— Every object has certain patterns with respect to their geographical architecture. Analyzing these patterns for tracking objects at real time is effective to contribute in artificial intelligence towards automation. There are various conventional methods have been introduced till now that recognize objects on the basis of patterns, Convolutional Neural Network (CNN) is one of them. But CNN is limited with certain objects because of analogous patterns of different objects. System confuses or does not work effectively when multiple objects are intended to recognize at real time. Tensorflow is most effective object detection API for identifying multiple objects at real time with high level of precision. It is used for both research and production at Google. Here the system uses tensorflow and Keras for machine learning along with Deep Neural Network to recognize different kind of objects in a single frame. Keras is an open source neural network library written in Python. It is capable of running on top of TensorFlow, Microsoft Cognitive Toolkit or Theano. Designed to enable fast experimentation with deep neural networks, it focuses on being user-friendly, modular, and extensible. Proposed system uses python for fine tuning the algorithm for object detection with different kind of packages such as pillow, pandas-datareader, scipy and jupyter. System is intended to classify the objects with high level of accuracy with quick response.

Keywords— Tensorflow, Keras, Object Detection, Pattern Recognition, CNN, DNN, Python.

I. INTRODUCTION

Object detection or classification is method of identifying objects on the basis of structural data which is a part of image processing in computer vision technologies. Detecting face, pedestrian etc is not a big deal but detecting multiple objects from a single frame at real time along with high level of accuracy and efficient prediction for unknown objects is hard. Tensorflow is a library through which an object can be detected with higher precision rate. In tensorflow an object model has been trained to detect the presence of object with good score or prediction rate. Fifty percent probability is considered as valid prediction but less than that may not entertain because of low prediction rate. Tensorflow has not been trained for every object, but a system can be designed for unknown object by its own dataset training. There are two models for input and output, input takes RGB image of expected 300*300 pixels for better classification. Output is in an array form and there are four arrays for location, classes, scores and detections. There are two methods through which a system can be designed either by using pre-trained data or self trained data. Accuracy is based on number of iterations and loss functions definitions. Higher score level declares higher accuracy.

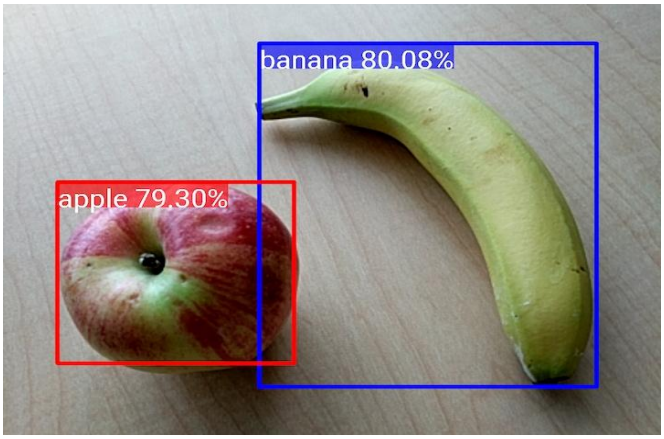


Fig. 1. Object Classification [1]

Fig. 1 shows object classification and their score level of probabilities and the prediction is correct and system efficiently recognized the objects and classify them in a single frame.

II. RELATED WORKS

A. Literature Survey

Davide Mulfari et al. [3] proposed a system whose main motive is to check how the proposed mobile system recognizes artifacts (in Messina city) is depicted in original image files (JPG format) taken from Google Image Search Engine Tanserflow based classification software runs on a minor Linux desktop machine, it calculates each visual data and after

processing, gives us a data structure that has ten text descriptions. Nitin R. Gavai et al. [4] proposed a system which is based MobileNets model on TensorFlow platform to retrain the flower category datasets, which can greatly minimize the time and space for flower classification compromising the accuracy slightly. MobileNet uses deep sense and makes sense each feedback channel applies a different filter. MobileNet. The model is built on intense intelligent intricacies which is complex as a factor converts a standard compound into a deep intelligent complexion and a 1x1 complexity is called dubious complexity.

After that the point war complexity applies to 1x1 complexity add output to depth intelligent complexity. In a standard both complexity are filtered and add input to make a different output set in one step. Diego Renan Bruno et al. [5] proposed a system which is based on traffic signal sign recognition at real time for driver assistance. System is able to classify different traffic signs (e.g. maximum speed allowed, stop, slow down, turn ahead, pedestrian). The training used in this work was implemented in the upper layer over CNN, which is based on the Inception Network. System uses an algorithm to detect traffic plates using slide window technique by giving an input image. Rasika Phadnis et al [6] proposed a system which is based on Tensorflow for recognizing objects and classify them. System tries to eliminate set alarms or reminders. It's the goal of making atmosphere of the future where all the actions of a person is kept in mind and it helps to maximize its efficiency. It will not only bring the way things are done but orders for discrepancies a better investigation will be done. Shaukat Hayat et al. [7] proposed a system which is based on CNN for recognizing objects at real time. The object recognition is the classic technique that is used for identifying an object effectively in the image. Especially the area of computer vision is expected to detect objects with the help of local facilities recognize more complex tasks ways to find out the objects. In the last decade, there has been continuous increase in the number of researchers from various types of topics such as education, industry, security agencies and even the general public has also attracted attention to its detection object aspects of detection and related recognition covered problems. Nicolas Diaz Salazar et al. [8] proposed a system which is based on Convolutional Neural Network. In this work, transfer learning techniques are used to create a computational device that recognizes objects at real time. As per pre-trained neural network, inception-V3 is used in the form of a feature extractor in the images and one on the other the softmax classifier is trained, it contains classes which are being recognized. This GPU was used with Tensorflow platform originally the OpenKeyl Library for Python and Use in Windows 10 of video camera and other devices.

III. Problem Identification

As per the survey takes place on various researches made in the field of real time object classification; most of the systems are based on CNN that intended to train a network in the reference of object patterns and features. System trails with the precision and processing time for decision making to classify objects at real time and action accordingly. Tensorflow is technique through which an object can be recognized at real time but classifying objects among various objects are difficult task. Multi-objects recognition in a single frame with high level of accuracy is bit challenge. A system is required to recognize objects at real time with high precision rate that met the technical configuration effectively. Needless to say, this algorithm can only understand the objects / sections which he has learned. In order to simplify things, we will focus on only two-tier (binary) classifier in this post. You might think that this is a very limited perception, but keep in mind that many popular object detectors (such as face detector and pedestrian detector) have a binary classifier under the hood. Like inside the face detector there is an image classifier that says that the patch of the image is face or background or not. Here the CNN based system acquired 90.12 % of accuracy or precision of recognizing and classifying objects. System has been tested over 5 different classes of wild animals and performance has been analyzed accordingly.

IV. Proposed Work & Implementation

The objective of this paper is to identify the multiple objects from a single frame at real time with high level of accuracy. System has been designed under Deep Neural Network training with pre-generated datasets. Keras has been used as loss function that enhances the prediction rate and proposed a better result in the field of object identification.

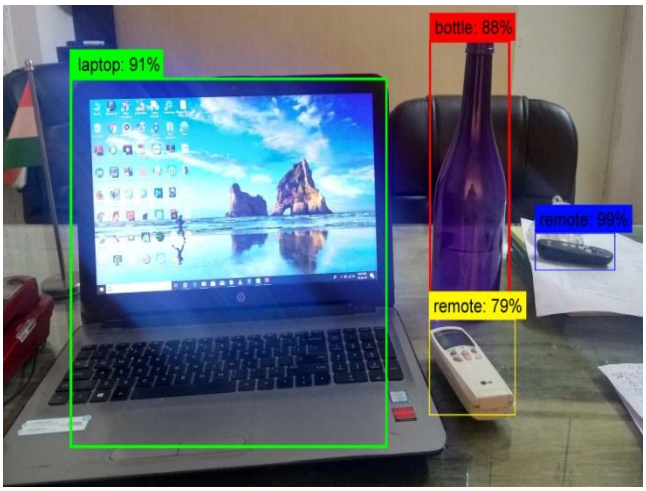


Fig. 2. Proposed API

Fig. 2 shows the proposed API of the system that recognized various objects from a single frame. An accurate machine learning model is capable of making and identifying many items in the same image remains a main challenge in computer vision. The TensorFlow Object Detection API is an open source framework built on top of TensorFlow which makes it easy to build, train and deploy an object detection model. Here the system is based on DNN (Deep Neural Network) and the training is done accordingly. Neura network is a training network where system has been trained for observation w.r.t. input and output data.

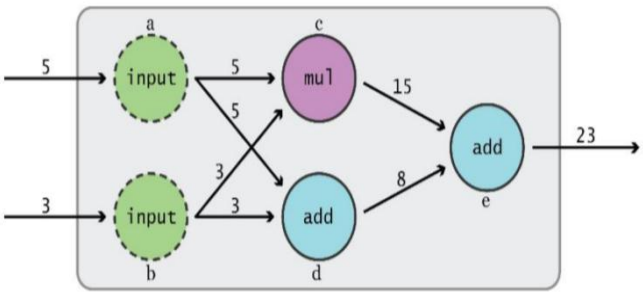


Fig. 3. Computational Neural Network [2]

Fig. 3 is a representation of computational neural network where mathematical observation has been done as per the best decision is awaited. Result is based on the training datasets and the loss function that backtracks results on the basis of hidden layers. Here the loss function is based on Keras API and Keras is a high-level API to build and train deep learning models. It's used for fast prototyping, advanced research, and production. Keras can works effectively with Tensorflow for obtaining high performance level.

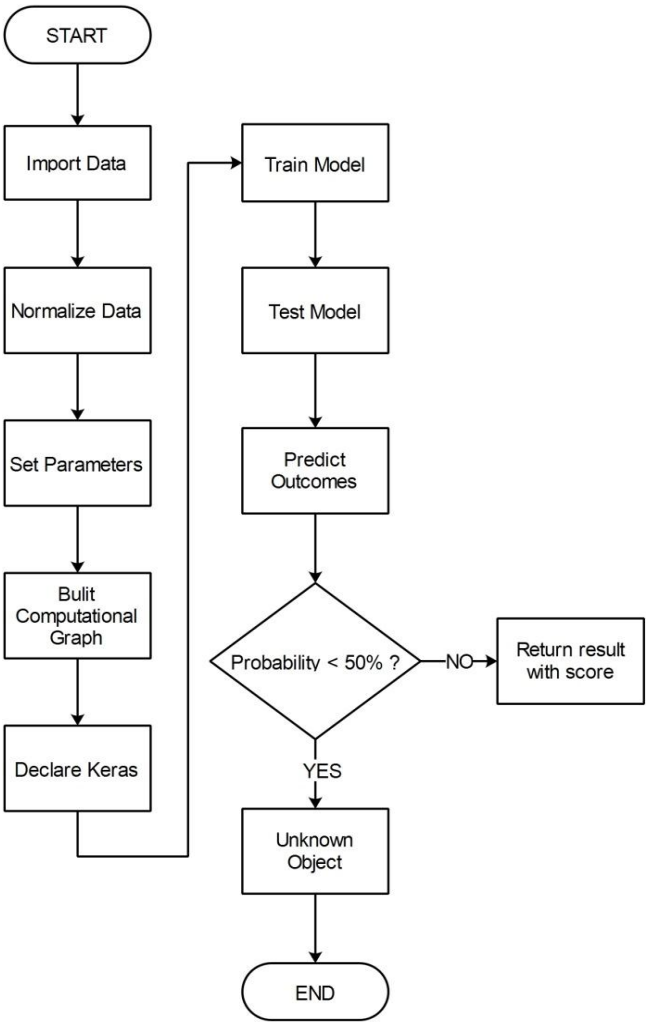


Fig. 4. Flow Chart

Flow chart represents the process model for recognizing objects with optimal prediction level. First of all data has to imported for execution and once the data imported it is required to normalize data as per the system compatibility then parameters are to be declared for Tensorflow variables. Our algorithm usually has a set of parameters that we keep constant during the whole process. For example, it can be number of iterations, learning rate or other specific parameters of our selection. It is considered a good form to start it together so that the reader or user can easily find them. After we have the data, and initialized our variables and placeholders, we have to define the model. This is done by building a computational graph. We tell Tensorflow what operations must be done on the variables and placeholders to arrive at our model predictions. After defining the model, we must be able to evaluate the output. This is where we declare the loss

function. The loss function is very important as it tells us how far off our predictions are from the actual values. Then train and test the model with DNN network for attaining better predictions. If obtained score is less than 50%, it means that object is unknown or out of training data else system returns the result accordingly with prediction score level.

A. ***Tensorflow & Keras Model Algorithm:***

Require: D as dataset, μ as mean, N is total no. of dataset, xi is an individual value, f as mantissa, e as biased exponent, s as sign bit, W & b as output parameters

Input: Datasets

Output: Probability Score

Step 1. Import or generate data

D = $\{x_n, y_n\}$ is a dataset

Step 2. Transform and normalize data by mean, standard deviation and float to integer

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2}$$

$$Where\ \mu\ is\ mean, N\ is\ total\ no.of\ dataset,$$
$$x_i\ is\ an\ individual\ value$$
$$(1-2s)\ \times\ (1+f)\ \times\ 2^e$$
$$s\ is\ the\ sign\ bit\ (0\ or\ 1), f\ is\ the\ mantissa$$
$$and\ e\ is\ the\ biased\ exponent$$

Step 3. Set Parameters W and b as tensorflow variables for result

Step 4. Built Computation Graph by initializing weights and biases

Step 5. Declare Keras as Loss Function

Step 6. Initialize and Train Model to better predict our data

Step 7. Evaluate the Model //Testing Phase

Step 8. Predict Outcomes & Declare Result

Step 9. End

Overall, algorithms are designed to be cyclic in TensorFlow. We set up this cycle as a computational graph and (1) feed in data through the placeholders, (2) calculate the output of the computational graph, (3) compare the output to the desired output with a loss function, (4) modify the model variables according to the automatic back propagation, and finally (5) repeat the process until a stopping criteria is met.

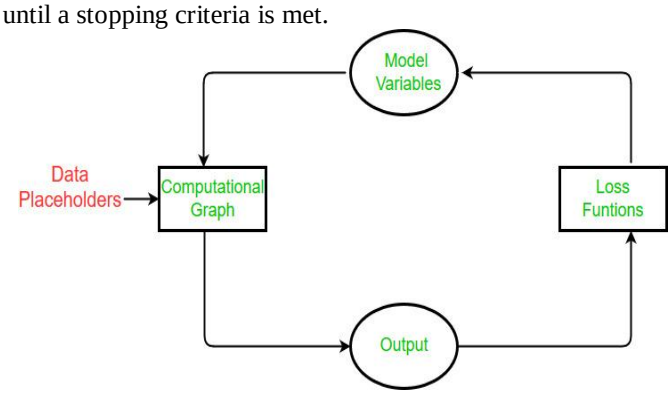


Fig. 5. Process Model

V. RESULT ANALYSIS

TABLE I. PROBABILITY SCORE LEVELS

Objects	Correct Score in %	Objects	Correct Score in %
1	88	26	97
2	85	27	92
3	91	28	88
4	99	29	90
5	92	30	88
6	95	31	99
7	87	32	95
8	85	33	89
9	98	34	93
10	99	35	89
11	81	36	91
12	87	37	85
13	90	38	99
14	99	39	97

15	91	40	85
16	90	41	99
17	99	42	90
18	94	43	99
19	90	44	96
20	97	45	97
21	89	46	90
22	90	47	91
23	99	48	83
24	86	49	90
25	91	50	93
Total			4597 (91.94)

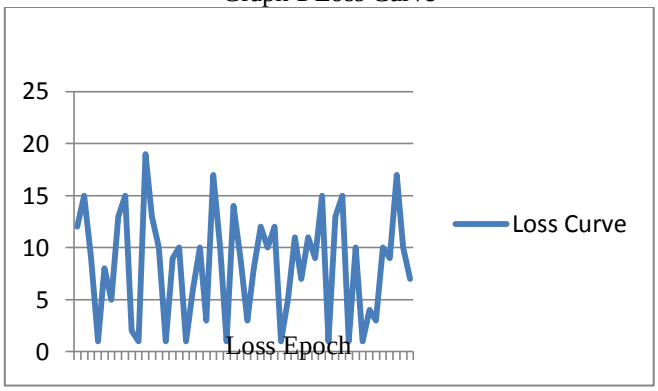
As the result table there are 50 objects have been observed and scores are recorded accordingly. All objects acquire valid score level and correctly recognized with distinct probabilities. As per the 50 prediction scores the accuracy has been measured as 91.94 %.

TABLE II. RESULT COMPARISON

Method	**Testing Accuracy in %**
BOW-Color with Max-Pooling	84.00
BOW-Color with Sum-Pooling	82.40
BOW-Gray with Max-Pooling	82.00
BOW-Gray with Sum-Pooling	81.40
HOG-BOW-Gray with Sum-Pooling	82.60
HOG-BOW-Gray with Max-Pooling	78.40
HOG-BOW-Color with Sum-Pooling	73.20
HOG-BOW-Color with Max-Pooling	63.60
CNN	90.12
DNN-Keras (Proposed)	91.94

As per the proposed result, the precision is bit higher than the previously proposed systems. They also tested for the same and acquired precision accordingly. Loss scores are also a part of testing phase where high loss attains less accuracy.

Graph 1 Loss Curve



VI. CONCLUSION & FUTURE SCOPE

The Keras based Tensorflow API acquires effective prediction rate as compare to the others. System is highly proficient to recognize and classifies multiple objects from a single frame with higher probability at real time. The proposed precision is 91.94 % which is obtained from 50 different scores of valid score patterns. Loss function does not degrade from 17 % as the highest loss score that predicted the objects correctly. So, the accuracy can be enhanced in future by minimizing loss score that also recognize objects effectively and precisely. It will play an important role in the field of computer vision technologies.

REFERENCES

[1] Tensorflow, "Object Detection", https://www.tensorflow.org/lite/models/object_detection/overview
[2] Chip Huyen, "Tensorflow Deep Learning Research", https://web.stanford.edu/class/cs20si/2017/lectures/slides_01.pdf
[3] D. Mulfari, A. Longo Minnolo and A. Puliafito, "Building TensorFlow Applications in Smart City Scenarios," *2017 IEEE International Conference on Smart Computing (SMARTCOMP)*, Hong Kong, 2017, pp. 1-5.
[4] N. R. Gavai, Y. A. Jakhade, S. A. Tribhuvan and R. Bhattad, "MobileNets for flower classification using TensorFlow," *2017 International Conference on Big Data, IoT and Data Science (BID)*, Pune, 2017, pp. 154-158.
[5] D. R. Bruno and F. S. Osorio, "Image classification system based on deep learning applied to the recognition of traffic signs for intelligent robotic vehicle navigation purposes," *2017 Latin American Robotics Symposium (LARS) and 2017 Brazilian Symposium on Robotics (SBR)*, Curitiba, 2017, pp. 1-6.
[6] Phadnis, Rasika & Mishra, Jaya & Bendale, Shruti. (2018). Objects Talk - Object Detection and Pattern Tracking Using TensorFlow. 1216-1219. 10.1109/ICICCT.2018.8473331.
[7] S. Hayat, S. Kun, Z. Tengtao, Y. Yu, T. Tu and Y. Du, "A Deep Learning Framework Using Convolutional Neural Network for Multi-Class Object Recognition," *2018 IEEE 3rd International Conference on Image, Vision and Computing (ICIVC)*, Chongqing, 2018, pp. 194-198.
[8] N. D. Salazar, J. Alfonso Lopez Sotelo and G. A. Salazar Gomez, "Application of Transfer Learning for Object Recognition Using Convolutional Neural Networks," *2018 IEEE 1st Colombian Conference on Applications in Computational Intelligence (ColCACI)*, Medellin, 2018, pp. 1-6.
[9] Zhou, B., Khosla, A., Lapedriza, A.., Oliva, A., Torralba, A.. Object detectors emerge in deep scene CNNs (2015).
[10] Brown, L. Deep learning with GPUs (2015)
[11] Loncomilla, P. Deep learning: Redes convolucionales (2016)
[12] The data science blog. An intuitive explanation of convolutional neural networks. Disponible en: https://goo.gl/KdqfLV.
[13] Szegedy,C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A. Going deeper with convolutions (2014)
[14] Alemi, A. Improving Inception and Image Classification in TensorFlow
[15] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z. Rethinking the Inception architecture for computer vision (2015)
[16] Lintz, N. Exploring computer vision (part II): Transfer learning (2016)
[17] https://indico.io/blog/exploring-computer-vision-transferlearning
[18] Tensorflow How to retrain inception´s final layer for new categories, https:// www.tensorflow.org/tutorials/image_retraining#bottlenecks
[19] Martinez-Martin, E. and A.P.d. Pobil, Object Detection and Recognition for Assistive Robots: Experimentation and Implementation. IEEE Robotics & Automation Magazine, 2017. 24(3): p. 123-138.
[20] Zhang, Y., H. Wang, and F. Xu. Object detection and recognition of intelligent service robot based on deep learning. in 2017 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM). 2017.
[21] Turaga, P., et al., Machine recognition of human activities: A survey. IEEE Transactions on Circuits and Systems for Video technology, 2008. 18(11): p. 1473-1488.
[22] Bhatnagar, S., et al., IITP at SemEval-2017 Task 5: An Ensemble of Deep Learning and Feature Based Models for Financial Sentiment Analysis. 2017.
[23] Szegedy, C., S. Ioffe, and V. Vanhoucke, Inception-v4, Inception- ResNet and the Impact of Residual Connections on Learning. 2016.
[24] Girshick, R., et al. Rich feature hierarchies for accurate object detection and semantic segmentation. in Proceedings of the IEEE conference on computer vision and pattern recognition. 2014.
[25] Krizhevsky, A., I. Sutskever, and G.E. Hinton, ImageNet classification with deep convolutional neural networks, in Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1. 2012, Curran Associates Inc.: Lake Tahoe, Nevada. p. 1097-1105.
[26] Fei-Fei, L., R. Fergus, and P. Perona, Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories. Computer vision and Image understanding, 2007. 106(1): p. 59-70.
[27] Li, F.-F., R. Fergus, and P. Perona. Learning Generative Visual Models from Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories. in 2004 Conference on Computer Vision and Pattern Recognition Workshop. 2004.
[28] Krizhevsky, A., I. Sutskever, and G.E. Hinton. Imagenet classification with deep convolutional neural networks. in Advances in neural information processing systems. 2012.