

Brand Logo Detection Using Convolutional Neural Network

Chandana N
Dept. Of ECE, GSSSIETW,
Karnataka, India

Chondamma P S
Dept. Of ECE, GSSSIETW,
Karnataka, India

Harshitha M
Dept. Of ECE, GSSSIETW,
Karnataka, India

Anusha M
Dept. Of ECE, GSSSIETW,
Karnataka, India

Dr. S. Padmashree
Professor, Dept. Of ECE, GSSSIETW,
Karnataka, India

Abstract— In this paper the method for logo detection using deep learning algorithms and python programming is being proposed. Pictures present an incredible opportunity for brands, not only do they have the potential to convey much more than text, they get shared more widely, clicked on more often, and are more easily digestible than text. A logo recognition system can therefore help brands get more insights from user generated content, optimize digital marketing strategy, and even protect trademarks against misuse. Our recognition pipeline is composed of a logo region proposal followed by a Convolutional Neural Network (CNN) specifically trained for logo classification, even if they are not precisely localized.

Keywords— Artificial intelligence, Machine learning, Deep learning, Convolutional Neural Network.

I. INTRODUCTION

AI[1][2] is an area of computer science that emphasizes the creation of intelligent machines that work and reacts like humans as shown in Fig 1.

- **Sense**—Identify and recognize meaningful objects or concepts in the midst of vast data.
- **Reason**—understand the larger context, and make a plan to achieve a goal. If the goal is to avoid a collision, the car must calculate the likelihood of a crash based on vehicle behaviour's, proximity, speed, and road conditions.
- **Act**—directly initiate the best course of action. Based on vehicle and traffic analysis, it may break, accelerate, or prepare safety mechanisms.
- **Adapt**—finally, we must be able to adapt algorithms at each phase based on experience, retraining them to be more intelligent.

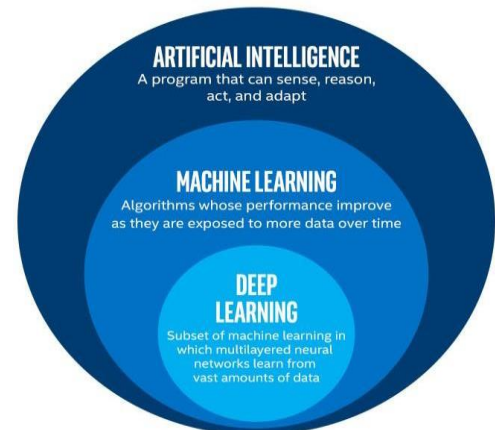


Fig. 1. Machine Learning

Machine learning[3] algorithms build a model from data as shown in Fig 2, which they can improve on as they are exposed to more data over time. There are four main types of machine learning: supervised, unsupervised, semi-supervised, and reinforcement learning

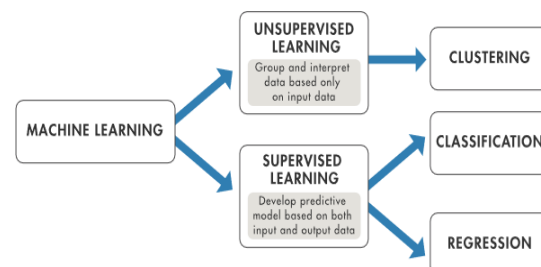


Fig. 2. Supervised learning

In supervised machine learning, the algorithm learns to identify data by processing and categorizing vast quantities of labelled data. In unsupervised machine learning [7], the algorithm identifies patterns and categories within large

amounts of unlabelled data—often much more quickly than a human brain could. A combination of the above is semi-supervised, used when there is a large amount of data but only some of it is labelled. Unsupervised learning techniques might be used to group and cluster the Unlabelled data, while supervised learning techniques can be used to predict labels for it.

Reinforcement learning uses simple reward data to train the machine on ideal behaviour within a specific context.

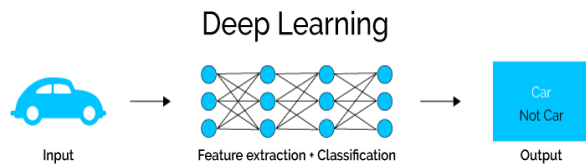


Fig. 3. Deep Learning

Deep Learning [4][5] is a type of Neural Network Algorithm that takes data as an input and process the data through some layers of the nonlinear transformation of the input data to compute the output as shown in Fig 3. This algorithm automatically grasps the relevant features required for the solution of the problem. It reduces the burden on the programmer to select the features explicitly. It can be used to solve supervised, unsupervised or semi-supervised type of challenges.

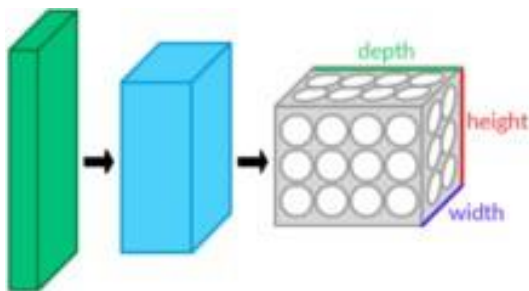


Fig. 4. Conventional Neural Networks

In machine learning, a Convolutional Neural Network (CNN, or ConvNet)[6][7][8] is a class of deep, feed-forward artificial neural networks that has successfully been applied to analysing visual imagery. A CNN consists of an input and an output layer, as well as multiple hidden layers. The hidden layers of a CNN typically consist of convolutional layers, pooling layers, fully connected layers and normalization layers. The layers of a CNN have neurons arranged in 3 dimensions: width, height and depth as shown in Fig. 4. The neurons inside a layer are connected to only a small region of the layer before it, called a receptive field. Distinct types of layers, both locally and completely connected, are stacked to form CNN architecture.

A **logo detection** method is employed to detect a few regions of interest (logo-patches), which likely contain the logo(s), in a document image.

Objectives--Detecting the brand logos in any image using convolutional neural network. Recognizing the brand logos in that image.

II. METHODOLOGY

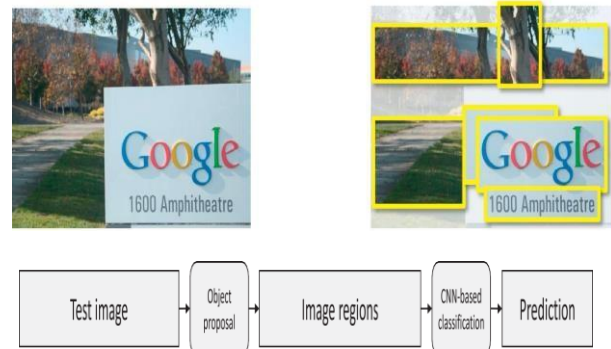


Fig. 5. Test Image

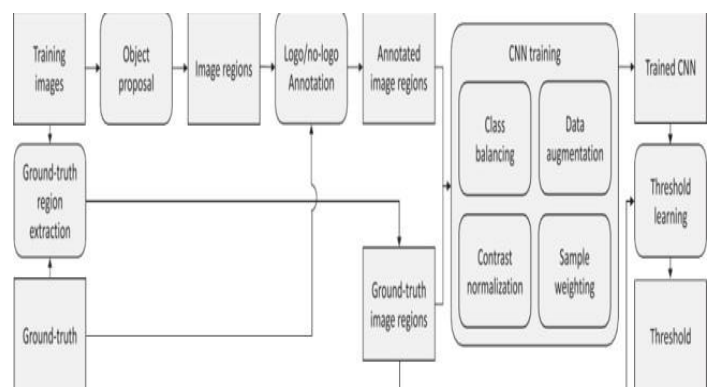


Fig. 6. Logo recognition Training framework

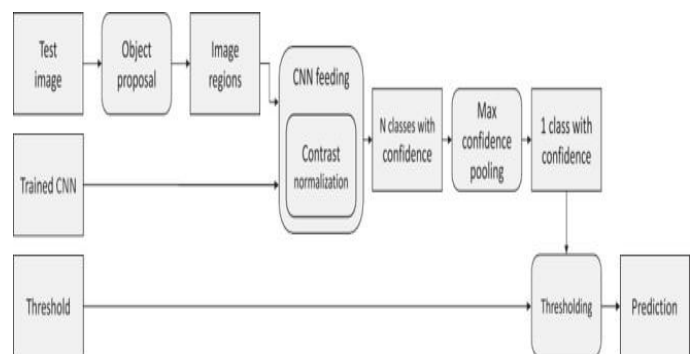


Fig. 7. Logo recognition testing framework

The proposed classification pipeline is illustrated in Fig. 5. Logos may appear in any image location with any orientation and scale, and more logos can coexist in the same image, for each image different object proposals are generated and these regions are more likely to contain a logo. These proposals are then cropped to a common size to match the input dimensions of the neural network and are propagated through a CNN specifically trained for logo recognition. In order to have performance as high as possible within this pipeline, an object proposal that is highly recall-oriented. For this reason, the CNN classifier is designed and trained to take into account the logo regions. The logo regions proposed may contain many false positives or only parts of actual logos. To

address these problems a training framework is proposed to investigate the influence on the final recognition performance of different implementation choices. In more detail, the training framework is reported in Fig. 6. The training data preparation is composed by two main parts:

- **Precise ground-truth logo annotations:** Given a set of training images and associated ground-truth specifying logo position and class, logo regions are first cropped and annotated with the ground-truth class. These regions are rectangular crops that completely contain logos but, due to the perspective of the image or the logo particular shape, may also contain part of the background.
- **Object-proposal logo annotations:** Since regions are automatically localized regions and they contain a logo, an object proposal algorithm is employed in the whole pipeline as shown in Fig.2.1. This algorithm is not applied only to the test images, but it is also run on the training images to extract regions that are more likely to contain a logo. Details about the particular algorithm used are given in the next subsection. Each object proposal in the training images is then labeled on the basis of its content and if the images overlap with a ground-truth logo region, it is then annotated with the corresponding class and with the Intersection over-Union (IoU) overlap ratio, otherwise it is labeled as background. Within our training framework investigations of precise ground-truth logo annotations alone or coupled with the object-proposal logo annotations.
- **Class balancing:** The logo classes are balanced by replicating the examples of classes with lower cardinality. Two different strategies are implemented: epoch-balancing, where classes are balanced in each training epoch, and batch-balancing, where classes are balanced in each training batch. The hypothesis is that this should prevent a classification bias of the CNN.
- **Data augmentation:** Training examples are augmented in number by generating random shifts of logo regions. The hypothesis is that this should make the CNN more robust to inaccurate logo localization at test time.
- **Contrast normalization:** Images are contrast-normalized by subtracting the mean and dividing by the standard deviation, which are extracted from the whole training set. The hypothesis is that this should make the CNN more robust to changes in the lighting and imaging conditions.
- **Sample weighting:** Positive instances are weighted on the basis of their overlap with ground-truth logo regions. The hypothesis is that this should make the CNN more confident on proposals highly overlapping with the ground truth logos.
- **Background class:** A background class is considered together with the logo classes. Background examples are not randomly selected, but are composed by the candidate regions generated by the object proposal algorithm on training images and that do not overlap with any logo. The hypothesis is that this should make the CNN more precise in discriminating logos and background class.

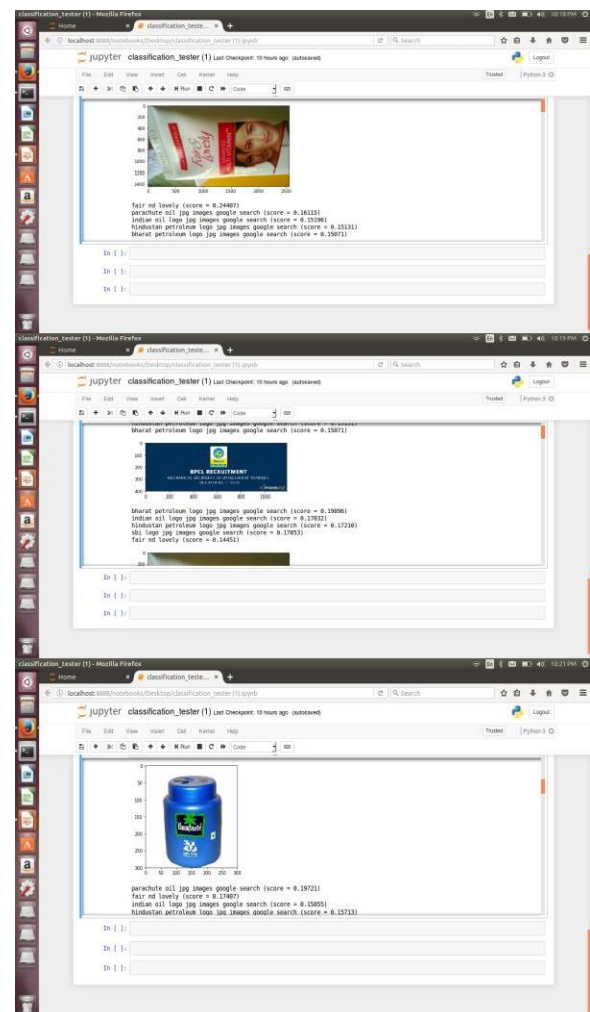
After the CNN is trained, a threshold is learned on top of the CNN predictions. If the CNN prediction with the highest confidence is below this threshold, the candidate region is

labeled as not being a logo, otherwise CNN prediction is left unchanged. The testing framework is reported in Fig.2.3. Given a test image, we extract the object proposals with the same algorithm used for training. We then perform contrast-normalization over each proposal (if enabled at training time), and feed them to the CNN. The CNN predictions on the proposals are max-pooled and the class identified with highest confidence (eventually including the background class) is selected. If the CNN confidence for a logo class is above the threshold that has been learned in training, the corresponding logo class is assigned to the image, otherwise the image is labeled as not containing any logo.

III. RESULT AND DISCUSSION

A. Steps To Find The Result

- Collect minimum of six hundred images for the desired classes
- Download the tensorflow modules
- Run the training code to train the system on all the images from a desired class
- Test given set of images for the detection of logos.
- Use inception V3[9] module for the purpose of training and testing.



IV. CONCLUSION

The problem faced in detecting a logo from a given image is solved by using a convolutional neural network, this is done by training the neural network to desired classes and later testing the given image for the purpose of detection.

REFERENCES

- [1] Celebi M. E, Kingravi H. A and Celiker F, "Accelerating image space transformations using numerical approximations" ,Image Processing (ICIP) 17th IEEE International Conference Publication Page(s): 1349-1352 2020.
- [2] A.K. Jain, "Fundamentals of Digital Image Processing", of detailed classification Upper Saddle River:Prentice-Hall Inc, 1989.
- [3] Shobha jayaram , Tanupriya choudhury , Praveen kumar
- [4] "Analysis of classification models based on cuisine prediction using machine learning", 2019 International conference of smart technologies for smart nation, pages:1485-1490
- [5] Yeong Jong Mo, Joongheon Kim, Jong-Kook Kim, Aziz Mohaisen, Woojoo Lee, "Performance of Deep Learning Computation with TensorFlow Software Library in GPU-Capable Multi-Core Computing Platforms", Proceedings of the IEEE International Workshop on Machine Intelligence and Learning (IWMIL), 4 July 2020.
- [6] Simone Bianco , Marco Buzzelli , Davide Mazzini , Raimondi Stettin , "Deep learning for logo recognition",University DISCo-Italy ,March 2021
- [7] F. Amira, A Bensaali, "Design and Implementation of Efficient Architectures for convolutional neural network Space Conversion", International Journal on Graphics Vision vol. 5, pp. 37-47, 2004.
- [8] Su-wen ZHANG1, Yong-hui ZHANG1, Jie YANG2 and Song-bin LI2, "Vehicle logo Recognition Based on Convolutional Neural Network with Multi-scale Parallel Layers",Proceedings of 2020 International conference on Computers,Mechatronics and Electronic Engineering(CMEE 2021)
- [9] Fatima Zahra Ouadiay; Hamza Bouftaih; El Houssine Bouyakhf; M. Majid Himmi, "Simultaneous object detection and localization using Convolutional neural network",Proceedings of 2018 International conference on Intellegent systems and computer vision (ISCV).
- [10] Xiaoling Xia; Cui Xu; Bing Nan."Inception V3 for flower classification",Proceedings of 2020 2nd International conference on Image ,Vision and Computing(ICIVC).