ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 7, Issue 3, pp: (95-99), Month: July - September 2019, Available at: www.researchpublish.com

Aircraft Detection in Satellite Images Using a Convolutional Neural Network

[1]Tanmay Kelkar, [2]Umang Sahastransu, [3]Zishen Thajudheen

***Abstract*: The project aimed at detecting aircrafts using a Convoluted Neural Network. The network was trained and tested which helped provide more accurate results. The inputs into the system were satellite images. This image was processed by being passed through the four layers of the network. The layers carried out the following tasks: i) Removal of Noise ii) Detection of Object iii) Feature Extraction of the object iv) Matrix Analysis so as to deliver the results. The proposed methodology gave the output in the form of the same satellite image along with the object being marked within squares. The functions written were divided to carry out the tasks of image identification and feature extraction. The Neural Network also needed to be trained prior to the detection process in order to ensure that the images would be detected with accuracy. This method in comparison with most other methods yielded better results.**

Keywords: Convolutional Neural Network (CNN).

I. INTRODUCTION

Object detection is one of the areas of computer vision that is growing very swiftly. It is observed that new algorithms are being developed every day and older algorithms are being modified and as such new object detection methods are on the rise. The main objective of this project is to implement an object detection algorithm such that the accuracy of the developed system will be as high as possible, at the same time ensuring that speed of detection is not compromised. Object detection involves a process wherein the model scans through the whole image and identifies a particular object in it. When it comes to object detection, a Convolutional Neural Network is tailored to identify objects in images and even for classification of images. Face identification and object detection are few of many areas in which the CNN is widely and extensively used.

II. MOTIVATION

Satellite images hold a large amount of data. Applications like environmental monitoring, spatial planning and natural resource management are among a few that have led to development of methods to extract remote sensor images. There is also a trend of increase in the number and sophistication of security applications, vehicle detection, and other urban applications that use remote sensor images. As there is an abundance of available images, machine learning approaches can be applied. Object recognition in images achieves high recognition rate when machine learning techniques are used. This is especially true in images which contain complex natural environment imagery. Satellite images can be considered to be in this classification of complex natural images.

III. METHODOLOGY

The aim of the project was to develop an aircraft detection system. First, the GPS coordinates of an area were required in a satellite image which contains aircrafts. The program that was developed downloaded the satellite data for that area. It then downloaded the dataset that was chosen to be used as the training data which contained images of aircraft along with images of natural objects. It used these aircrafts as positive training samples and the background, such as roads and trees as negative feedback. After the training data was gathered and the neural network was trained, it was used to detect various aircrafts consisting of different sizes. The steps could be stated as:

- Step 1: Input the datasets, both positive and negative, through the CNN to train it;
- Step 2: Input the satellite image at minimum distance of 1cm=100m (on the satellite image) into the CNN;
- Step 3: Run the detection algorithm to identify an object in the image based on the trained dataset;
- Step 4: Mark every identified object with a polygon to highlight it in the image.

IV. OVERVIEW OF CNN

A convolutional neural network is branch of deep learning which is used to analyze visual depiction of imagery. It consists of one or more convolutional layers which are followed by one or more fully connected layers as in a standard neural network with the advantage that they have fewer parameters for the same number of hidden layers. CNNs, like neural networks, are made up of neurons with learnable weights and biases. Each neuron receives several inputs, takes a weighted sum over them, pass it through an activation function and responds with an output. The input to the convolutional layer is a m x m x r image where m is the height and width of the image and r is the number of channels. For example, an RGB image will have r=3. The convolutional layer consists of k filters which are of size n x n x q where n is of a smaller order than that of the image and q is generally taken as the same as the number of channels r. The filter size determines the various features that are extracted by the convolutional process. The feature maps that result are convolved with the image to produce k feature maps of size m-n+1. Each feature map is then passed onto pooling functions and are made nonlinear using a Rectified Linear Unit function. After the convolutional layers there may be one or more fully connected layers which are used to map the extracted features to objects.

Convolutional layer:

Convolution is the first layer to extract features from an input image. Convolution preserves the relationship between pixels by learning image features using small squares of input data. It is a mathematical operation that takes two inputs such as image matrix and a filter or kernel. Convolution of an image with different filters can perform operations such as edge detection, blur and sharpen by applying filters.

The following equations can be used to calculate the exact size of the convolution output for an input with the size of (width = W, height = H) and a Filter with the size of (width =*Fw*, height =*Fh*):

Output width= $\frac{W-F_w+2P}{S_w}+1$

Output height= $\frac{H-F_h+2P}{S_h}+1$

Where *Sw* and *Sh* are horizontal and vertical stride of the convolution, respectively, and P is the amount of zero padding added to the border of the image.

Strides:

Stride is the number of pixels shifts over the input matrix. When the stride is 1 then we move the filters by 1 pixel at a time. When the stride is 2 then we move the filters by 2 pixels at a time and so on.

Padding:

Sometimes filter does not fit perfectly in the input image. In that case there are two options:

- Pad the picture with zeros (zero-padding) so that it fits.
- Drop the part of the image where the filter did not fit. This is called valid padding which keeps only valid part of the image.

Non-Linearity layer (ReLU):

ReLU stands for Rectified Linear Unit for a non-linear operation. The output is:

f(x)= max(0,x)

ReLU's purpose is to introduce non-linearity in the CNN. Since, the real-world data would want the CNN to learn would be non-negative linear values. There are other nonlinear functions such as tanh or sigmoid can also be used instead of ReLU. Most of the data scientists uses ReLU since performance wise ReLU is better than other two.

Pooling Layer:

Pooling layers section would reduce the number of parameters when the images are too large. Spatial pooling also called subsampling or down sampling which reduces the dimensionality of each map but retains the important information. Spatial pooling can be of different types:

- Max Pooling
- Average Pooling
- Sum Pooling

Max pooling take the largest element from the rectified feature map. Taking the largest element could also take the average pooling. Sum of all elements in the feature map call as sum pooling.

Fully Connected Layer:

The layer is called as FC layer in which the matrix is flattened into vector and feed it into a fully connected layer like neural network. In the above diagram, feature map matrix will be converted as vector (x1, x2, x3, …). With the fully connected layers, these features are combined together to create a model. Finally, there is an activation function such as SoftMax or sigmoid to classify the outputs as cat, dog, car, truck etc.

Backpropagation:

Neurons in CNN share weights in which every neuron has an individual weight vector. This process of sharing the weights reduces the number of trainable weights thus introducing sparsity. Using this weight sharing strategy, neurons are able to perform convolutions on data with the convolutional filter being formed by the weights. When this is followed by a pooling operation, which is a form of down-sampling, the size of the representation becomes smaller and leads to a reduction in the amount of computation and parameters in the network.

V. EXPERIMENTAL DATASET

The training process requires sample images, on the basis of which the network can identify objects. The larger the number of samples, the better the network becomes at identifying a particular object. Since obtaining a large number of such images is a tedious task, the PlanesNet dataset was been used. PlanesNet is a labelled training dataset consisting of image chips extracted from Planet satellite imagery. This consists of thousands of 20 x 20 pixel RGB images labelled as either having a plane or not having a plane. This constitutes the positive and negative image samples that are required for the training process. There are a variety of datasets available on the internet but since aircrafts were being detected in this system, the PlanesNet dataset was used and was obtained from the official website of PlanesNet.

VI. EVALUATION METRICS

In this project, the outputs that were obtained from the different inputs given to the system were compared with reality to check whether the evaluation metrics were satisfied. The evaluation metrics that were chosen were based on the confusion matrix defined as:

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 7, Issue 3, pp: (84-94), Month: July - September 2019, Available at: www.researchpublish.com

- *Confusion Matrix:* This is a matrix which describes the complete performance of the model as an output.
- *True Positives:* The cases in which the prediction was YES and the actual output was also YES.
- *True Negatives:* The cases in which the prediction was NO and the actual output was YES.
- *False Positives:* The cases in which the prediction was YES and the actual output was NO.
- *False Negatives:* The cases in which the prediction was NO and the actual output was also NO.

$$Accuracy = \frac{True\ Positives + False\ Negatives}{Total\ no.of\ Samples}$$

Performance analysis of the results put in confusion matrices for different number of input images given in three separate runs of the neural network:

No. of Blocks:1118	Expected: NO	Expected: YES
Actual: NO	1031	6
Actual: YES	2	79

Accuracy=99.28%

No. of Blocks:537	Expected: NO	Expected: YES
Actual: NO	503	4
Actual: YES	4	26

Accuracy=98.51%

No. of Blocks:1118	Expected: NO	Expected: YES
Actual: NO	546	17
Actual: YES	12	16

Accuracy=95.09%

VII. FUTURE WORKS

A lot of techniques which include features such as histogram of oriented gradient, local binary pattern, scale-invariant feature transform, etc. have been used to improve the performance of object detection but the success in doing so has been very limited, confining the detection mostly in simple environments such as roads and the difficult part of detecting small objects such as various vehicles still exists. But a technique based on Deep convolutional neural networks (DNNs) has shown remarkable results in image classification databases as it can learn different features of a particular object from the training data automatically. The deep convolutional neural network (DNN) being a feature learning architecture is being extensively used in many object recognitions tasks. The DNN uses the convolution layers and the max-pooling layers. The hidden layers and the output layer combine the extracted features for classification.

VIII. LIMITATIONS

- Training and detection time increase as the image size increases and neural network requires a very powerful GPU to do the computations.
- False identification of object can occur in any given satellite image.

REFERENCES

[1] Alexander Toshev, Ben Tasker, Kostas Daniilidis."Shape based object detection via Boundary Structure Segmentation", Google Research ,September 2003,pp. 251-303

[2] Inad A. Aljarrah, Ahmed S. Ghorab, Ismail M. Khater."Object Recognition System using Template Matching Based on Signature and Principal Component Analysis", The Society of Digital Information and Wireless Communication, November 2012, pp. 115- 160

[3] Ben Weber. "Generic Object Detection using AdaBoost", Department of Computer Science University of California, September 2013,pp. 220-301

[4] Szegedy, Christian, Alexander Toshev, and Dumitru Erhan. "Deep Neural Networks for Object Detection." In Advances in Neural Information Processing Systems,October 2013,pp. 2553–61

[5] Erhan, Dumitru, Christian Szegedy, Alexander Toshev, and Dragomir Anguelov. "Scalable Object Detection Using Deep Neural Networks." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, February 2014,pp. 2147–54.

[6] Ouyang, W et al. "DeepID-Net: Deformable Deep Convolutional Neural Networks for Object Detection." In IEEE Conference on Computer Vision and Pattern Recognition (CVPR),October 2015,pp. 2403–2412.

[7] Aleksej Avramović, Igor Ševo."Convolutional Neural Network Based Automatic Object Detection on Aerial Images", IEEE Geoscience and Remote Sensing Letters,October 2016,pp. 50-170

[8] Kevin Murphy, Antonio Torralba, Daniel Eaton, William Freeman."Object detection and localization using local and global features", Department of Computer Science, University of British Columbia,August 2016,pp. 231-287

[9] Subarna Tripathi, Gokce Dane, Byeongkeun Kang, Vasudev Bhaskaran, Truong Nguyen. "LCDet: Low-Complexity Fully-Convolutional Neural Networks for Object Detection in Embedded Systems",IEEE Conference on Computer Vision and Pattern Recognition Workshops,January 2017,pp. 128-190

[10] Bernardo De Oliviera, Carlos Martins."Fast and Lightweight Object Detection Network:Detection and Recognition on Resource Constrained Devices", IEEE Geoscience and Remote Sensing Letters March 2017,pp. 156-201

[11] Wang Zhiqiang, Liu Jun. "A review of object detection based on convolutional neural network", 36th Chinese Control Conference,December 2017,pp. 179-261

[12] Ajeet Ram Pathaka, Manjusha Pandeya , Siddharth Rautaray. "Application of Deep Learning for Object Detection", International Conference on Computational Intelligence and Data Science ,January 2018,pp. 123-170