International Journal of Computer Trends and Technology
ISSN: 2231-2803 / https://doi.org/10.14445/22312803/IJCTT-V68I2P103
Volume 68 Issue 2, 17-21, February 2020

Original Article

Ship Detection from Satellite Imagery in Deep Learning: Using Sequential Algorithm

Kodanda Dhar Naik[1], Manisha Rautaray[2], Shivam Sharma[3], Sourav Mohapatra[4], Subhashree Dash [5], Abhishek Parida[6]

Department-CSE & University-BPUT, Parala Maharaja Engineering College, Berhampur, Odisha, India

Received Date: 02 January 2020
Revised Date: 11 February 2020
Accepted Date: 14 February 2020

Abstract — *Ship detection is an inherent process supporting tasks such as fishery management, ship search, marine traffic monitoring and control, and helps in the prevention of illegal activities. So far, sea and shore monitoring has been carried out by ship patrols and aircraft, along with sea vessel detection from data from space-borne platforms.* ***While*** *investigating the state of the art methods used for ship detection from different platforms using optical images, we found a significant problem with the occurrence of a ship wake. This phenomenon may prohibit correct detection of ship location and results in overestimating the ship size as the ship, and its wake is often considered as being part of the same object in an image, or wakes are distinguished as a separate ship due to their possible similar brightness compared with sea vessel. In order to reduce the impact of ship wakes, we investigated the behaviour of images in different colour spaces to provide data with little or almost no trace of ship wake. An object of interest was detected through the use of image segmentation. The applied method uses edge detection based on the gradient magnitude calculation.*

Keywords *- Deep Learning, Remote Sensing Convolutional Neural Network, Keras Model, Sequential Algorithm.*

I. INTRODUCTION

Satellite imaging, or Remote Sensing, is the scanning of the Earth by satellite or high-flying aircraft in order to obtain information about it[5]. There are many different satellites scanning the Earth, each with its own unique purpose. For example, Landsat, MODIS, Sentinel, ASTER, Meteosat.[6] While humans can perceive only a small portion of the EM spectrum (visible light), satellite sensors can use other types, like infrared light, ultraviolet light, or even microwaves. Satellite images are useful because different surfaces and objects can be identified by the way they react to radiation. For instance, smooth surfaces, such as roads, reflect almost all of the energy which comes at them in a single direction. This is called specular reflection. Meanwhile, rough surfaces, such as trees, reflect energy in all directions. This is called diffuse reflection. Satellite images are required by meteorologists to predict weather forecasts, any natural phenomenon or to detect any kind of activity under surveillance. There are many different types of satellite images. Satellite imagery depicts the Earth's surface at various spectral, temporal, radiometric, and increasingly detailed spatial resolutions, as is determined by each collection system's sensing device and the orbital path of its reconnaissance platform.

In Remote Sensing, we have 4 types of resolutions for satellite imagery:

- Spatial resolution- Spatial resolution is a term that refers to the number of pixels utilized in the construction of a digital image.
- Spectral resolution- Spectral resolution describes the ability of a sensor to define fine wavelength intervals.
- Temporal resolution- Temporal resolution is defined as the amount of time needed to revisit and acquire data for the exact same location.
- Radiometric resolution- Radiometric resolution refers to how much information is in a pixel and is expressed in units of bits.

II. WHY USE SATELLITE IMAGES FOR SHIP DETECTION?

While humans can perceive only a small portion of the EM spectrum (visible light), satellite sensors can use other types, like infrared light, ultraviolet light, or even microwaves. Satellite images are useful because different surfaces and objects can be identified by the way they react to radiation. A ship is made up of steel and other metals, which causes it to reflect a lot of energy that is incident upon it. This is called specular reflection. Meanwhile, other surfaces, such as peripheral water, reflect a very little amount of energy, which helps in the detection of a ship inside water.

III. WORKING MODEL

A. Convolutional Neural Network

Convolution is a specialized kind of linear operation. Convolution networks are simply neural networks that use convolution in place of general matrix multiplication in at least one of their layers [2]. A convolutional neural network consists of an input and output layer, as well as multiple hidden layers. Keras model comes under a Convolutional neural network.

B. Keras Model

Keras is an open-source neural network library written in python. It is capable of running on top of TensorFlow.It contains different types of neural network building blocks [1], which supported the sequential model which works on the layers such as –

- objectives
- Activation function
- Optimizers

C. Sequential Model

The sequential model is only a very young concept and has transformed deep learning in a huge way. Basically, a lot of data we receive or generate today are in the form of sequences, for example, sentences or numbers, speech etc. These are all very natural for us to understand, but a network of neurons used to fail to understand it. Let alone predict an output. We can define a sequential model as a model of networks that takes sequential data as input and produces single fixed-size output, which is now used as input in another layer of the model. The primary goal of sequential modelling is to use the data (which is in a sequential format) and make it usable for the network, which has practical applications like speech to text conversion, chatbox skill development, DNA sequencing etc. We can visualise it as a linear stack of layers that passes a list of layer instances to the constructor. Models have all these layers and functions. Layers are the building blocks of the sequential model. Layers are functions that contain some internal states called boots. These weights are either trainable/non-trainable.

D. Algorithm For Sequential Model

Step 1:-Keras is imported along with the other python modules, libraries and packages.

Step 2:- In this step the data is loaded into the program.

Step 3:-In this step thee image is reshaped after the deduction of the image in the previous step.

Step 4:-The plotting of different kinds are done here with the help of libraries.

Step 5:-Data pre-processing is done here.

Step 6:- Comparison of standard classifiers is done and box plotted graphs are produced.

Step 7:-Other helper functions are defined to ease computation.

Step 8:- The applied convolutional neural networks are assessed.

Fig. 1 Sequential algorithm

E. Layers of Sequential Model

- *Lambda layer:*-It has no internal state. It is an anti-rectifier. It exists so that arbitrary tensor flow functions can be used when constructing the Sequential and Functional API model. Lambda layers are best suited for simple operations or quick experimentations.
- *Dense layer:* - A dense layer represents a matrix-vector multiplication. The values in the matrix are the trainable parameters, which get updated during backpropagation.
- *Convolutional layer:*-Convolutional layers are the major building blocks used in a convolutional neural network. This is the simple application of a filter to an input that results in inactivation. The result is highly specific features that can be detected anywhere in the input image. It allows the hierarchical decomposition of inputs.
- *Max pool layer:*-Maxpool layer summarizes the output of the convolutional layer and reduces the spatial dimensions of input. Maxpooling is a pooling operation that selects the maximum element from the region of the feature map covered by the filter.
- *Dropout layer:*-A dropout layer is used for regularisation, where you randomly set some of the dimensions of your input vector to be zero with probability keep_prob. It doesn't contain a trainable parameter.[3]

F. Flow Chart of Sequential Model

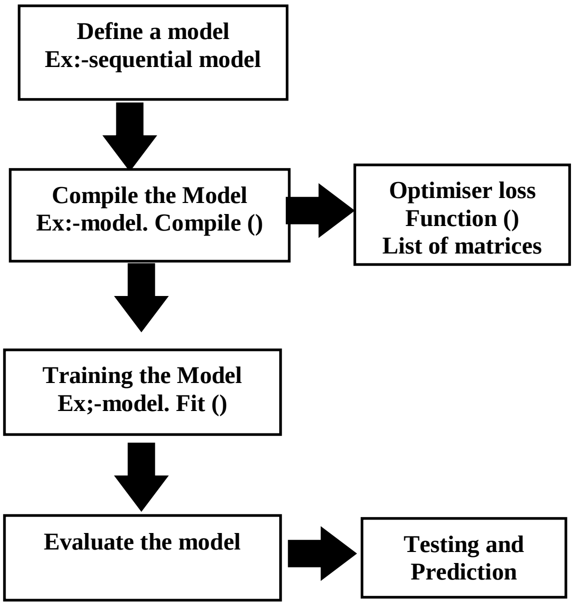

Fig. 2 flow chart of a sequential algorithm

A. Parameters of Input Layers

- FILTER SIZE[32]:- This is the of the output dimension (i.e., this is the output filters in the convolution)
- KERNEL_SIZE [3, 3]:- This specifies the height and width of the 2d convolution window.
- ACTIVATION [RELU]:- We select an activation function also called non-linearity to be used by our neural network. RELU is the most common activation function used today. Other variations are leaky RELU and ELU.
- POOL SIZE [2, 2]:- This specifies the size of the Max pooling window.
- INPUT SHAPE [80, 80, 3] - It defines the height weight and channel dimension which we give to the input layer.
- PADDING: - It allows us to design a deeper network. Without paddings, dimensions are shrunk in the last layer one of “valid” or “same”).

B. Compilation

Before training a model, we need to configure the learning process, which is done by the compile method. It receives three arguments.

- Optimizer: - This could be the string identifier of an existing optimiser (Stochastic Gradient Descent) or an instance of the optimiser class.
- Loss function: - This is the objective that the model will try to minimise. It can be the string identifier of an existing loss function (categorical _cross entropy or MSE), or it can be an objective function.
- A list of metrics: - For any classification problem, you will want to set this to metrics= ['accuracy']. A metric could be the string identifier of an existing metric or a custom metric function. See metrics.

C. Training

Keras models are trained on NumPy arrays of input data and labels. For training, we use 3 functions:

- Fit function:-It trains the model with a fixed number of epochs.
- Fit generator:-It is a bit complex as it takes a generator instead of a NumPy array. It is often used for large databases.
- The train on batch function:-It allows us to do a single gradient update over on batch of samples.

Then we use our data & model together to go ahead and fit the model.

D. Testing

Testing is the most important part of the evaluation.

- The image from the training phase is accepted as an input in this phase.
- Using the cutting function, the image is divided into multiple small images, which are used in the next step.
- These images are now provided with X and Y coordinates values.
- The images which have ships in them are then appended, and output is plotted and localized.

IV. GRAPHICAL REPRESENTATION

In the portion mentioned below, the code snippet is used to take a certain small space bounded by the given values of coordinates x and this space is then searched for ships, and if the accuracy of more than 90% is seen, the ship (or portion of it) is detected.

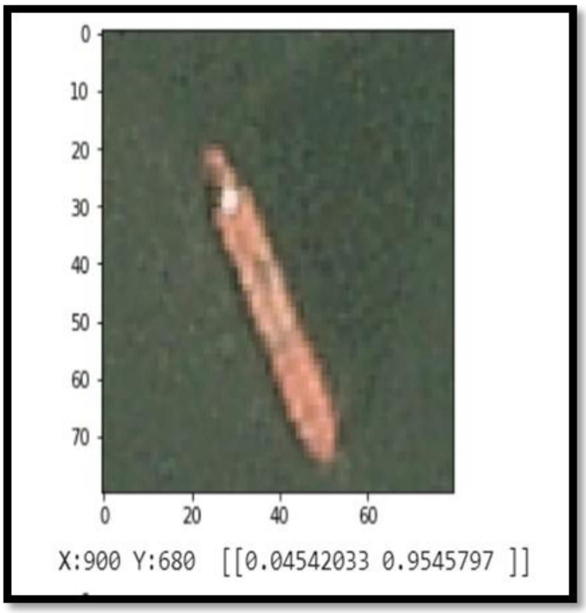

Fig. 3 Graphical Representation of a ship

V. RESULT

When we processed satellite image data using the Sequential Model of Convolutional Neural Networks (CNN), we could detect ships and locate them.

a) Ship Detection

In Ship detection, we detect ships and other ship like objects present in the satellite images.

Fig. 4 Ship Detection

b) Ship Localization-While performing Ship Localisation, we determined each and every part of the ship in its location area.

Fig. 5 Ship localiztion

VI. CONCLUSION

Ship detection is a very important aspect of maritime surveillance. Ships play a very critical role in a country's development and well-being. Ships help in import-export, crude oil extraction, guarding our navy etc. For all these purposes, we require techniques like ship detection from satellite images.

In this paper, we use Convolutional Neural Networks (CNN) to detect and localise ships from Ship images. We presented a system designed for automatic ship detection on high spatial resolution optical satellite imagery. We have chosen the sequential network model, which is a part of the Keras model in CNN, which we have done using three steps that is compilation, training and testing of the datasets. In this, we have taken the inputs, and we applied the loss functions that decrease the loss value and increase

the accuracy. Then we train the dataset by fit function. After that, it goes for the testing process, and we detect the ships and localize them. The approach gets better and better with each subsequent iteration. The procedure is completely unsupervised and takes only a few minutes, depending on hardware characteristics. This technique for ship detection was augmented by pre-processing and post-processing to reduce both the computation time and the number of false alarms. The pre-processing involves cloud masking and +local contrast enhancement of image tiles. Most of these false alarms are due to local wind turbulences, which increases background noise. Some other false alarms are related to confusions between moving targets with characteristic ship wakes and other naturally produced linear features such as internal waves or wind-waves crests. Some large boats with their wakes were considered as several small ship targets and hence resulted in multiple detections that augmented the total number of false positives. A great deal of effort is being undertaken to improve validation procedures and control efficiency by i) introducing information from other maritime monitoring systems, such as VMS, ii) cross-cuing to other sensors, such as SAR sensor, for obtaining or confirming detections. Further experiments are required in order to render the system fully operational. For instance, it would be interesting to adapt the algorithm to various ship sizes and types and to test in on very high spatial resolution optical imagery (sub metric pixel resolution).

VII. FUTURE SCOPE

As the Keras sequential model does not support generalising the data. It is only applicable for certain data sets, and it includes only a few features of deep learning. Thus the algorithm is unable to detect the ships efficiently. Future research work includes incorporating location features into our classification system in order to obtain the ship position within optical aerial images. Other sensors, such as SAR, could additionally be evaluated for situations in which it is not possible to use visible spectrum imagery, as occurs at night. Different sensors could also be combined in a multimodal scenario. In addition to classifying whether or not an image contains a ship, the exact location of the ship detected could also be extracted by means of saliency estimation methods. We also plan to add more images to the MASERATI dataset to make it larger and build better models. In order to speed up this process, semi-supervised strategies could be considered.

REFERENCES

[1] Vitali Burachorok, Keras for search ships in a satellite image. (2017).

[2] Eijaz Allibhai, Data Science & Android Development, Building a Convolutional Neural Network in Kera. (2018).

[3] Apurv Verma, ML Research @Georgia Tech, Dense & Dropout layer. (2017).

[4] Mathew Mayo, Kdnuggets, SAS Global, Washington DC. (2018).

[5] Weiss J, Luo R, Welch R, Automatic detection of ship tracks in satellite imagery. In Proceedings of the IEEE International Geoscience and Remote Sensing (IGARSS 97), Remote Sensing A Scientific Vision Crisp, D. The State-of-the-Art in Ship Detection in Synthetic Aperture Radar Imagery. Australian Government, Department of Defense: Canberra, Australia. (2004) 115.

[6] Greidanus H, Kourti N, Findings of the DECLIMS project Detection and Classification of Marine Traffic from Space. In Proceedings of the SEASAR Advances in SAR Oceanography from ENVISAT and ERS Missions, Frascati, Italy. (2006) 23–26.

[7] Marino A, Sanjuan-Ferrer M.J, Hajnsek I, Ouchi K, Ship Detection with Spectral Analysis of Synthetic Aperture Radar: A Comparison of New and Well-Known Algorithms. Remote Sens. (2015).

[8] LeCun Y, Bengio Y, Hinton G, Deep learning Nature. 521 (2015) 436 - 444.