Journal of Physics: Conference Series

PAPER • OPEN ACCESS

Deep learning-based object detection and geographic coordinate estimation system for GeoTiff imagery

Deep learning-based object detection and geographic coordinate estimation system for GeoTiff imagery

B M Pratama[1,3], D Gunawan[2] and R A G Gultom[1,4]

[1] Department of Sensing Technology, Indonesia Defense University, Bogor, Indonesia
[2] Department of Electrical Engineering, University of Indonesia, Depok, Indonesia
[3] Department of Physics, Kalimantan Institute of Technology, Balikpapan, Indonesia
[4] Ministry of Defense of Republic of Indonesia, Jakarta, Indonesia

Email: bmpratama@staff.itk.ac.id

Abstract. A deep learning-based system has been created to autonomously analyze GeoTiff aerial imagery in order to retrieve information about objects type and their geographic coordinates. This research focuses on applying a Convolutional Neural Network (CNN) to detect objects and estimate the geographic coordinate of airplanes, ships and cars in those images. The system prototype was tested to measure the accuracy and precision for object detection. Furthermore, a Mean Absolute Error (MAE) analysis is done to the system to measure object coordinate estimation performance. The accuracy and precision for object detection of the system prototype are 81,05% and 93,29%, respectively. The system has MAE values which vary from 0,000012° to 0,000034° for object coordinate estimation.

1. Introduction

The development of technology in the era of Industry 4.0 is characterized by the implementation of artificial intelligence to optimize the current system [1]. Technological advancement in the field of defense technology must also adapts to the characteristic of Industry 4.0 by implementing artificial intelligence. Implementation of artificial intelligence in the field of defense has been done to detect the presence of certain object in aerial and satellite imagery such as cars [2,3], airplanes [4,5] and ships [6].

For defense purpose, especially for satellite or drone-based surveillance, object detection only is not enough. It is also important to determine the geographic coordinate of the detected object in aerial or satellite imagery-based surveillance. The geographic coordinate is needed to let the authority to intercept the detected object if needed. This research proposed a system for detecting and estimating coordinate of specific objects in GeoTiff images using Convolutional Neural Network (CNN). CNN has been proven resulting a good performance to detect objects from aerial and satellite imagery [2,3,5]. The CNN used in this research is modified from You Only Look Once (YOLO) model [7] to make it able to simultaneously detect airplanes, cars, and ships inside of GeoTiff imagery and estimate the object coordinate.

2. Research Method

2.1. CNN

CNN used in this research is based on YOLO model [7] which has been modified to fit the needs of this research. The advantage of using YOLO model is the rapidity of the model because the whole image is fed to the CNN input instead of sliding window or regional-based techniques [7]. The process of object detection in YOLO model is shown in Figure 1. The whole image is simultaneously processed by the CNN after the image is divided into several regions. Furthermore, the CNN predicts the presence of object in each region.

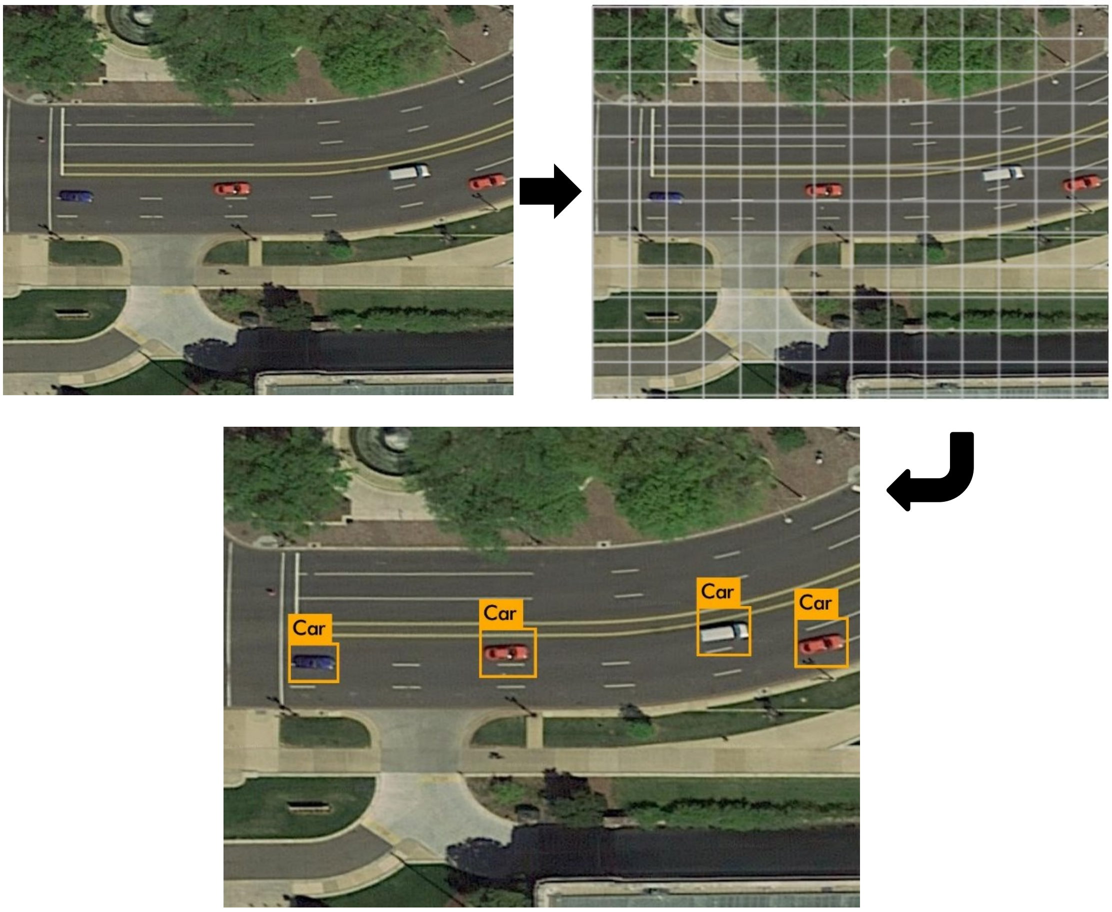

Figure 1. Object detection process in YOLO model.

Each detected object is overlaid with a bounding box. Every bounding box contains four parameters: x, y, w and h. The parameters x and y are the pixel coordinate of the detected object relatively to image width and height. The w and h are respectively the width and height of the detected object relatively to the image width and height. Therefore, x, y, w and h values are normalized between 0 and 1.

The YOLO model CNN is modified using C++ programming language to fit the needs of this research. The CNN used in this research consists of 31 layers. 16 convolutional layers and 3 fully-connected layers are used instead of the original version which has 24 convolutional layers and 2 connected layers [7]. Those layers are shown in Figure 2. Preliminary experiments resulted that the CNN best performance was achieved while using 928 × 928-pixel of input image. Therefore, every image is resized to 928 × 928-pixel before being fed to the CNN input. The CNN was trained for 7500 iterations using 165 training images which are containing 350 airplanes, 352 ships and 854 cars. Cars are composing most of the training images because car detection is expected to be the most difficult object to detect. Cars have more various shapes, color and background than airplanes and ships.

Training images are obtained from Google Earth. Samples of training images used are shown in Figure 3. The initial learning rate for the training is 0.001. The learning rate is reduced to 0.0001 and 0.00001 for after the 4000^{th} and 5500^{th} iteration. The training process was done using Nvidia GTX 750 Graphic Processing Unit (GPU).

```
layer     filters    size              input                output
    0 conv     16  3 x 3 / 1   928 x 928 x   3   ->   928 x 928 x  16
    1 max          2 x 2 / 2   928 x 928 x  16   ->   464 x 464 x  16
    2 conv     32  3 x 3 / 1   464 x 464 x  16   ->   464 x 464 x  32
    3 max          2 x 2 / 2   464 x 464 x  32   ->   232 x 232 x  32
    4 conv     64  3 x 3 / 1   232 x 232 x  32   ->   232 x 232 x  64
    5 max          2 x 2 / 2   232 x 232 x  64   ->   116 x 116 x  64
    6 conv    128  3 x 3 / 1   116 x 116 x  64   ->   116 x 116 x 128
    7 max          2 x 2 / 2   116 x 116 x 128   ->    58 x  58 x 128
    8 conv    256  3 x 3 / 1    58 x  58 x 128   ->    58 x  58 x 256
    9 max          2 x 2 / 2    58 x  58 x 256   ->    29 x  29 x 256
   10 conv    512  3 x 3 / 1    29 x  29 x 256   ->    29 x  29 x 512
   11 max          2 x 2 / 1    29 x  29 x 512   ->    29 x  29 x 512
   12 conv   1024  3 x 3 / 1    29 x  29 x 512   ->    29 x  29 x1024
   13 conv    256  1 x 1 / 1    29 x  29 x1024   ->    29 x  29 x 256
   14 conv    512  3 x 3 / 1    29 x  29 x 256   ->    29 x  29 x 512
   15 conv     24  1 x 1 / 1    29 x  29 x 512   ->    29 x  29 x  24
   16 yolo
   17 route  13
   18 conv    128  1 x 1 / 1    29 x  29 x 256   ->    29 x  29 x 128
   19 upsample           2x     29 x  29 x 128   ->    58 x  58 x 128
   20 route  19 8
   21 conv    256  3 x 3 / 1    58 x  58 x 384   ->    58 x  58 x 256
   22 conv     24  1 x 1 / 1    58 x  58 x 256   ->    58 x  58 x  24
   23 yolo
   24 route  21
   25 conv    128  1 x 1 / 1    58 x  58 x 256   ->    58 x  58 x 128
   26 upsample           2x     58 x  58 x 128   ->   116 x 116 x 128
   27 route  26 6
   28 conv    128  3 x 3 / 1   116 x 116 x 256   ->   116 x 116 x 128
   29 conv     24  1 x 1 / 1   116 x 116 x 128   ->   116 x 116 x  24
   30 yolo
```

Figure 2. The CNN architecture used in this research.

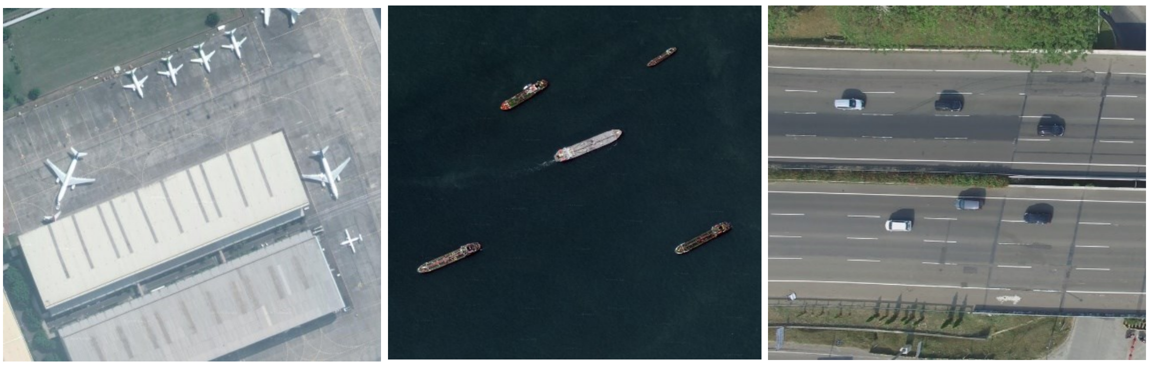

Figure 3. Some images used for CNN training.

2.2. Geographic Coordinate Estimation

The object geographic coordinate is written in decimal degree format. Geographic coordinate for n^{th} bounding box of the detected object is obtained by using equation (1) and (2) as follow:

$$long_n = lim_{long} + (L_{long} \times x_n) \tag{1}$$

$$lat_n = lim_{lat} + \{L_{lat} \times (1 - y_n)\} \tag{2}$$

where $long_n$ and lat_n are the longitude and latitude of the detected object, lim_{long} and lim_{lat} are the geographic coordinate (longitude and latitude) of the bottom left pixel of the image, L_{long} and L_{lat} are the length of longitude and latitude of the image, x_n and y_n are the pixel coordinate of the detected object. Both x_n and y_n are obtained from the CNN while $long_n$, lat_n, lim_{long}, lim_{lat}, L_{long} and L_{lat} are obtained from the GeoTiff image metadata.

3. Experimental Results

3.1. CNN Testing

The trained CNN is tested using 60 images containing 151 cars, 91 airplanes and 80 ships. Images are collected from Google Earth in GeoTiff format. CNN detection results are categorized into True Positive (TP), False Positive (FP) and Negative (N). The detection is stated as TP when the object is detected and classified correctly. When the object is detected but not classified correctly, the detection is denoted as FP. Some objects which are not detected by the CNN is classified as N. Samples of TP, FP and N images are shown in Figure 4. The system uses red, green and yellow bounding boxes to mark detected airplanes, ships and cars, respectively.

a. b. c.

Figure 4. Examples of detection results: a. TP b. FP c. N

CNN performance is measured by calculating the accuracy and precision of the system. Accuracy is the amount of closeness of the measurement result to a quantity's true value, while precision is the degree of closeness of some measurement results in the same condition [8]. Therefore, the system is considered to be more valid if the system is accurate and precise. The percentage of accuracy and precision of the system is measured using equation (3) and (4) as follow:

$$A = \frac{TP}{TP+FP+N} \times 100\% \tag{3}$$

$$P = \frac{TP}{TP+FP} \times 100\% \tag{4}$$

where A and P are the percentage of accuracy and precision of the system.

Table 1. CNN detection results.

Object Type	*TP*	*FP*	*N*	*A*	*P*
Airplane	81	9	10	81.00%	90.00%
Ship	60	4	20	71.43%	93.75%
Car	137	7	15	86.16%	95.14%
Overall	278	20	45	81.05%	93.29%

The CNN detection result is summarized in Table 1. Airplane, ship and car detection resulted 81.00%, 71.43% and 86.16% accuracy, respectively. Some of the detection results are shown in Figure 5. Airplane and car detections tend to have better accuracy than the ship detection. This is because airplanes can be easily distinguished from the background. The background is usually having a darker color than the airplane itself. Car detection has the best performance than the others because the training set contains more car images than airplane or ship images.

Figure 5. Examples of detection results. Airplanes, ships and cars are marked with red, green and yellow bounding boxes, respectively.

Negative detection could be found mostly in ship and car detection. Ocean surface usually reflects sunlight making detection harder because the ships are not clearly visible. Some other objects are not detected because they usually disguised by the environment. Few tactical airplanes found in the images are painted with certain colours to let them camouflage themselves. Ships and cars with relatively same color with the ocean or roads are difficult to detect. The other objects are not detected because they are partially covered by the environment such as some cars are covered by shadow or trees. False positive detections are usually happened on an object which is visually similar with airplanes, ships or cars. Some negative and false positive detections are shown in Figure 6.

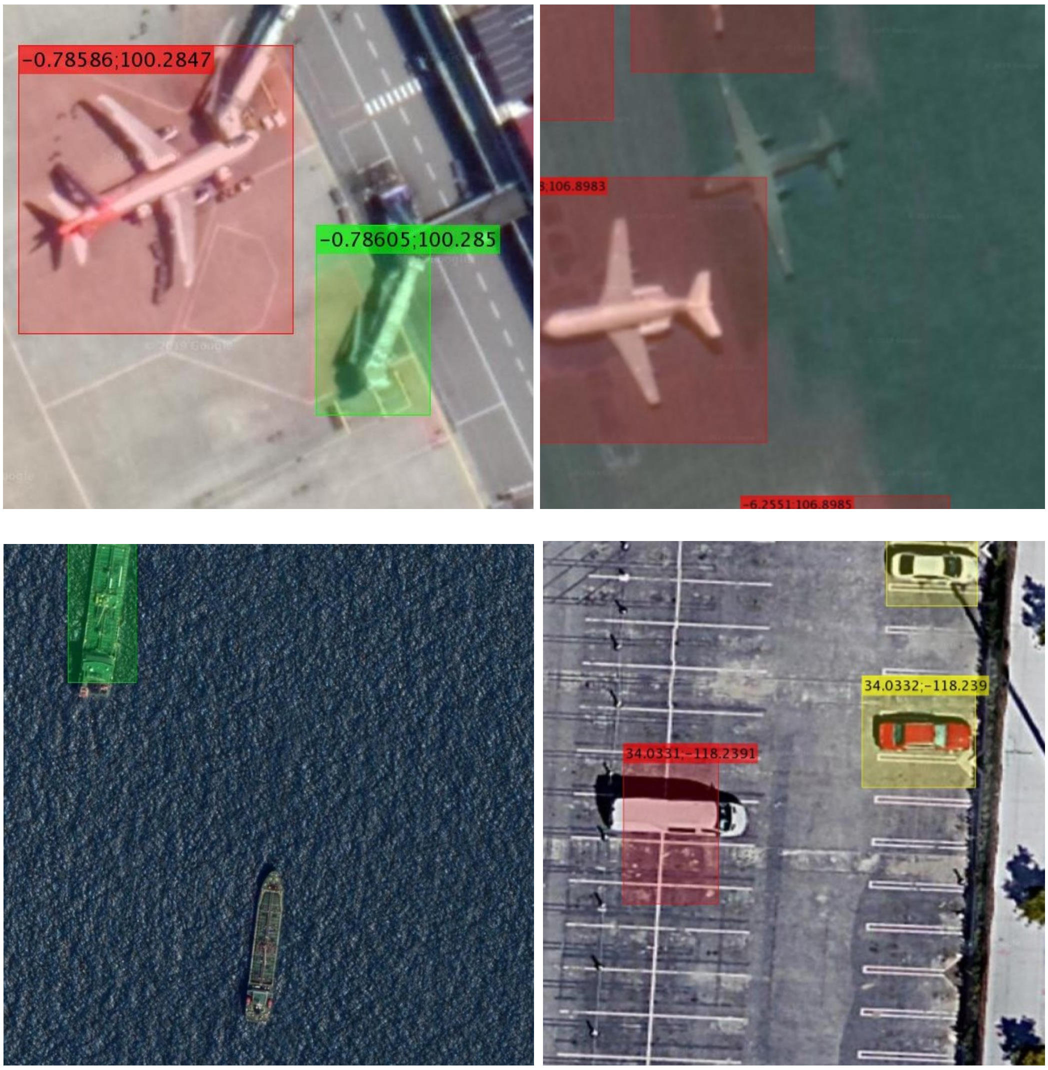

Figure 6. Examples of negative and false positive detections.

3.2. Geographic Coordinate Estimation Testing

Geographic coordinate estimation performance is measured using Mean Absolute Error (MAE) analysis. MAE is chosen because it is considered better for measuring average model-performance error rather than Root Mean Squared Error (RMSE) [9].

MAE is calculated using equation (5) and (6) to measure the error of longitude and latitude of the geographic coordinate estimation:

$$MAE_x = \frac{1}{n}\sum|x_n - x_n'| \tag{5}$$

$$MAE_y = \frac{1}{n}\sum|y_n - y_n'| \tag{6}$$

where MAE_x and MAE_y are the MAE of the longitude and latitude, respectively. Both MAE_x and MAE_y units are in degree. They are indicating the average of coordinate estimation inaccuracy. The number of detected objects is denoted with n. Predicted longitude and latitude are denoted with x_n and y_n while the real longitude and latitude are denoted with x_n' and y_n'. Predicted longitude and latitude are displayed to each bounding box of the detected object as shown in Figure 7. Real longitude and latitude are the geographic coordinate of the object's centroid. The value of x_n' and y_n' are manually obtained using QGIS application for every object.

Figure 7. Coordinate estimation is displayed on bounding boxes.

Geographic coordinate estimation results are summarized in Table 2. Coordinate estimation for cars has the best performance with $MAE_x$and MAE_y value of 0.000012° and 0.000014°. This because cars are relatively smaller than airplanes and ships so that the coordinate estimation is not deviating much from the object centroid. Airplanes and ships relatively same size resulted the relatively same value of MAE between 0.000027° up to 0.000034°. In general, larger object generates larger inaccuracy of geographic coordinate estimation.

Table 2. Geographic coordinate estimation results.

Object Type	$\sum\lvert x_n - x_n'\rvert$	$\sum\lvert y_n - y_n'\rvert$	n	MAE_x	MAE_y
Airplane	0.002418°	0.002409°	81	0.000030°	0.000030°
Ship	0.001602°	0.002044°	60	0.000027°	0.000034°
Car	0.001738°	0.001902°	137	0.000012°	0.000014°

4. Comparison to Previous Researches

The CNN accuracy in this research is generally lower than previous proposed methods. This is because the detected object is classified into three classes while the other researches classify the detected object into only one class. The main difference to previous proposed methods is the ability of estimating object's geographic coordinate in GeoTiff satellite imagery which could not be found in the previous researches. The comparison of the proposed method to previous researches is shown in Table 3.

Table 3. Comparison to previous researches.

Model	**Objectives**	**Method**	**Accuracy**
Model 1 [3].	• Detecting cars.	CNN	96.719%
Model 2 [4].	• Detecting airplanes.	*Visual saliency + symmetry detection*	93%

Model	Objectives	Method	Accuracy
Model 3 [5].	• Detecting airplanes.	CNN	97.5%
Model 4 [6].	• Detecting ships.	*Local binary patterns + Support Vector Machine*	89.22%
This model	• Detecting airplanes, cars and ships. • Estimating the geographic coordinate of the detected objects	CNN	81.05%

5. Conclusion
A deep learning-based object detection and geographic coordinate estimation system has been created. The proposed system is applicable for GeoTiff imagery. The detection system is based on YOLO model CNN. The CNN output is processed to calculate the geographic coordinate of the detected objects. The overall detection accuracy and precision are 81.05% and 93.29%, respectively. The geographic coordinate estimation test resulted varying MAE from 0.000012^{o} up to 0.000034^{o} depending on the size of the detected object. Coordinate estimation on smaller objects tends to have lower error than coordinate estimation on larger objects.

6. References

[1] Vaidya S, Ambad P and Bhosle S 2018 Industry 4.0 – A Glimpse *Procedia Manuf.* **20** 233–8

[2] Chen X, Xiang S, Liu C-L and Pan C-H 2014 Vehicle Detection in Satellite Images by Hybrid Deep Convolutional Neural Networks *IEEE Geosci. Remote Sens. Lett.* **11** 1797–801

[3] Jiang Q, Cao L, Cheng M, Wang C and Li J 2015 Deep neural networks-based vehicle detection in satellite images *2015 International Symposium on Bioelectronics and Bioinformatics (ISBB)* 2015 International Symposium on Bioelectronics and Bioinformatics (ISBB) (Beijing, China: IEEE) pp 184–7

[4] Li W, Xiang S, Wang H and Pan C 2011 Robust airplane detection in satellite images *2011 18th IEEE International Conference on Image Processing* 2011 18th IEEE International Conference on Image Processing (ICIP 2011) (Brussels, Belgium: IEEE) pp 2821–4

[5] Radovic M, Adarkwa O and Wang Q 2017 Object Recognition in Aerial Images Using Convolutional Neural Networks *J. Imaging* **3** 21

[6] Yang F, Xu Q, Gao F and Hu L 2015 Ship detection from optical satellite images based on visual search mechanism *2015 IEEE International Geoscience and Remote Sensing Symposium*

(IGARSS) IGARSS 2015 - 2015 IEEE International Geoscience and Remote Sensing Symposium (Milan, Italy: IEEE) pp 3679–82

[7] Redmon J, Divvala S, Girshick R and Farhadi A 2016 You Only Look Once: Unified, Real-Time Object Detection *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (Las Vegas, NV, USA: IEEE) pp 779–88

[8] Patterson J and Gibson A 2017 *Deep learning: A practitioner's approach* (O'Reilly Media, Inc.)

[9] Willmott C and Matsuura K 2005 Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance *Clim. Res.* **30** 79–82