Journal of Computer Science and Applications.
ISSN 2231-1270 Volume 15, Number 1 (2023), pp. 11-19
© International Research Publication House
https://dx.doi.org/10.37624/JCSA/15.1.2023.11-19

MultiScale Object Detection in Remote Sensing Images using Deep Learning

Ch. Radhika, B SiriPriya, D. Nikhila, Bhavana Reddy and D. Shivani

Dept. of CSE, G. Narayanamma Institute of Technology and Science (for women)
Hyderabad, India
ch.radhika@gnits.ac.in, siripriyab2@gmail.com, donthireddynikhila@gmail.com
bhavanareddykanthala@gmail.com, dasarishivani2@gmail.com

Abstract

With a rapid development in aerial technology, applications of Remote Sensing Images (RSI) have become more diverse. Remote sensing object detection is a difficult task due to complicated background, variations in the scales of the objects and proximity between objects of same scale. RSI's are commonly captured from satellites with wide views, which leads to large-scale images.

The proposed model detects the objects at different scales. Feature Extraction and providing additional information about the object is done using Residual Neural Network101 (ResNet101) and ZFNet. Further, single scale and multiscale object detection is implemented using You Only Look Once (YOLOV5) and Faster Region based Convolutional Neural Network (Faster RCNN). A comparative study is done on all these techniques to evaluate the performance measures like Mean Average Precision and Accuracy.

Key words: Remote Sensing Images, Feature Extraction, Multi Scale Object Detection

I. INTRODUCTION

Remote sensing object detection (RSOD) is the most researched topic in Remote Sensing Images (RSI). It locates the object regions of interest and classifies the multi objects present. Remote Sensing Object Detection still remains as a challenge because of complex scenarios and variations in the scales of the objects [1]. Remote Sensing Images are captured from satellites having wide views, which lead to the variations of scales in images and complex background. These are the main obstacles for object detection in Remote Sensing Images. They have many applications which include hazard response, urban monitoring, traffic control and many more [2]. Noise can be removed from grayscale and color photographs with a lot of techniques [4].

The algorithms that have been effective in natural scene images are not adapted to aerial images taken in wide view. Convolutional Neural Networks are used based on their performance with the natural images. The object detection algorithms can be classified into one stage and two stage object detector methods. The one stage method performs in a one step process whereas the two stages perform region extraction and classifying bounding boxes.

Faster RCNN involves the design of region proposal network. The one stage object detector method constitutes YOLO [11], Retina Net [14]. YOLO works by dividing the image into several cells through a single network.

Feature Pyramid Network (FPN) has been incorporated for multiscale object detection but these can only address the imbalances present at the feature level. To address the above issues and improve the detection accuracy and reduce computation time and adaptive network is proposed which consist of feature extraction techniques that contain additional information about the object and object detection algorithm that give a superior accuracy.

II. Literature Survey

Machine Learning has been incorporated in the past times for remote sensing object detection. Any machine learning algorithm included feature extraction, selection and classification. Initially, Random Forest has been employed, later region based Convolution Neural Network (RCNN) has been used due to its feature extraction capability [12]. In order to enhance the detection performance feature fusion strategy is used [13]. Further, to deal with the complex background and noises leading to low resolutions linear regression has been employed [14]. Examining the differences and similarities between the concepts. Along with this scenes categorization and rectification has been used [15].

The methods that are used to remove noise and deal with complicated background images include: Histogram of Oriented Gradients (HOG) feature, latent support vector machine (SVM) to train deformable mixture models [2]. The object detection methods such as YOLOV3, Retina Net have used Feature Pyramid Network for multi scale object detection by coalesce feature maps [9]. Anchor Boxes design has been widely used with algorithms such as You Only Look Once, Single Shot Detector. Object detection algorithms have to predict bounding box with the class it belongs to and confidence scores [14]. Feature Pyramid Network employs the techniques of up sampling and element wise summation to solve multiscale object detection. The algorithms YOLOV3 and Retina Net employ this strategy. [9]. Remote sensing object detection algorithms have been enhanced with context enhanced modules. An example that uses this is CAD Net model.

Detectors such as Faster RCNN uses anchor based detectors [15]. The anchor boxes can be defined as a sliding window that can be categorized into positive or negative in addition with refining the bounding box prediction. Feature maps of the CNNs are used by the anchor boxes so as to resist the feature computation that is repetitive and rapidly increasing the detection speed [17].

Faster RCNN was responsible for the design of the anchor boxes and one stage detectors such as YOLOV2 has been widely used in the modern detectors.

Object detection is still considered to be challenging in the field of computer vision, where there is a need to predict the bounding box with the class label and confidence score associated with the object in the image [18].

III. METHODOLOGY

The following step by step procedure implements the proposed model for detecting the objects in remote sensing images.

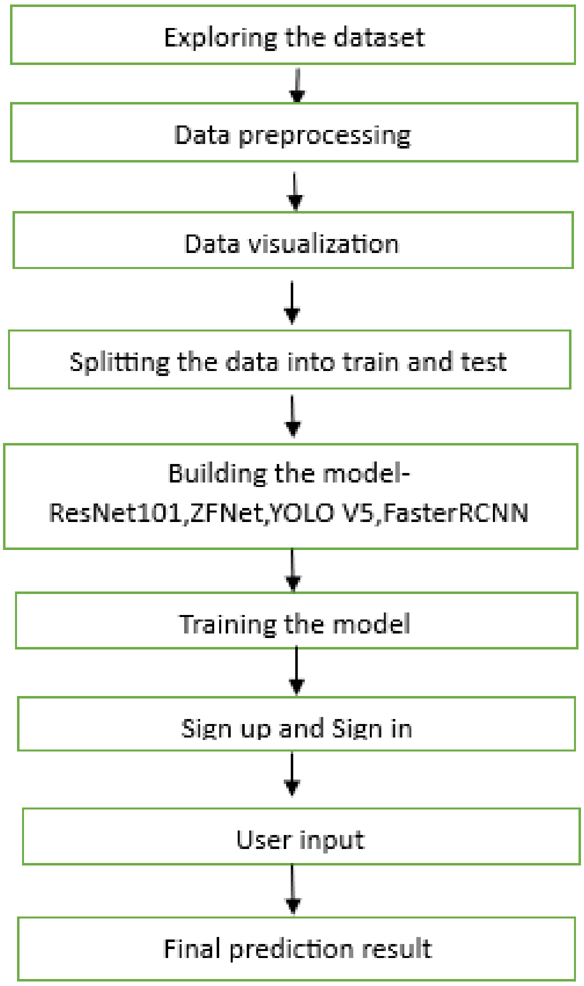


Fig 1. Flow of proposed system

A. Data collection:

Data was collected from aerial satellite images dataset.

B. Preprocessing:

Images have been preprocessed. Data augmentation is done to increase the data for custom object detection.

C. Feature Extraction:

The features from images are extracted using Residual Networks 101 and Zeiler and Fergus Net.

D. You Only Look once (YOLOV5):

YOLOV5 processes the images by dividing it into different components to increase the performance of the model.

E. Faster Region based Convolution Neural Network (Faster RCNN):

Faster RCNN is used for the purpose of object detection based on deep convolutional network. An image is given as a input and confidence scores, bounding boxes are given as a output.

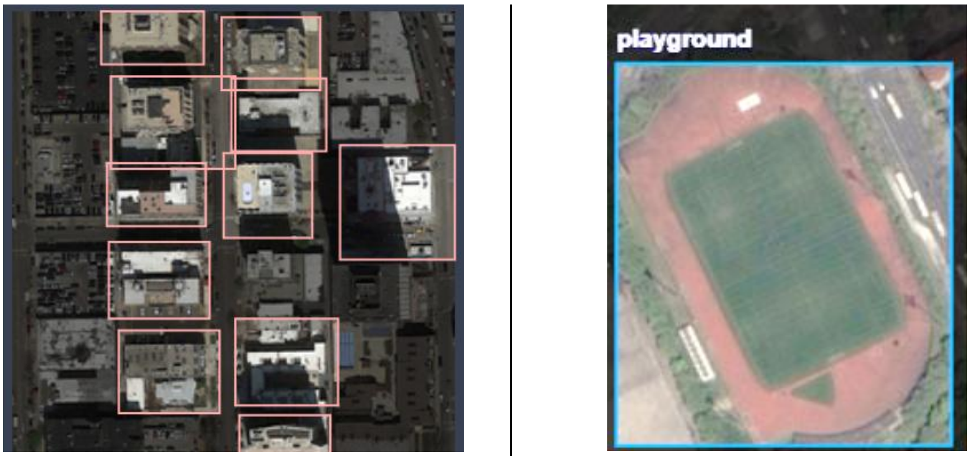
F. Object detection:

The model is used to predict single scale and multi scale objects after training using YOLOV5.

IV. IMPLEMENTATION

A. Dataset

The Satellite images dataset is used to implement the proposed system. Farmlands, factories, playgrounds, residential areas, airplane, and parking lots are the different sorts of classes. There are 1272 images in the entire dataset. Manual Annotation is done using RoboFlow tool and Data Labelling is done by incorporating Boundary Box.



(a) (b)

Figure 2. Data Annotation (a) Boundary Box (b) Data Labelling.

B. Algorithms:

YOLOV5: The single stage object detector is used to detect the objects of different scales in a remote sensing image with custom dataset and provides an output of bounding box around the object with the confidence score. CSPNet can be employed in order to extract the features of the image. YOLOV5 uses pytorch implementation which overcomes the challenges of the darknet frameworks.

Faster RCNN: This model is used for comparison with YOLOV5. It uses deep convolutional neural network and it uses the regions of interest pooling layer for the extraction of feature vectors. It appears to be a single, unified network that provides the output with class probabilities and accuracy.

V. RESULTS

When the YOLOV5 algorithm is used in the process, the images are trained and tested, and the final result is obtained with the bounding box around the object and corresponding confidence score.

A. Performance Measures:

For analyzing the YOLOV5 and Faster RCNN Models, the Mean Average Precision (mAP) and accuracy are looked over to figure out how well the model works.

Mean Average Precision (mAP):

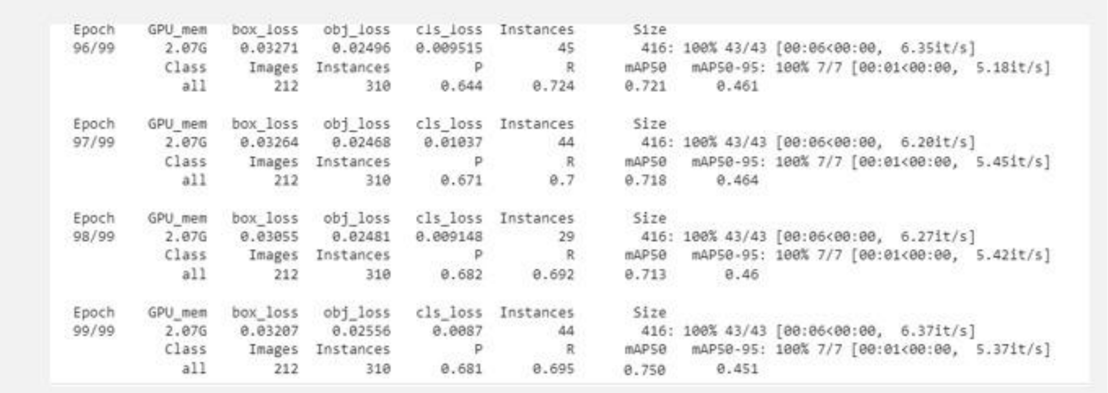
The widely used metric for the purpose of object detection and is applied to all the classes are Average Precision and Mean Average Precision. If the value of the mAP is high, then the detection performance is also higher. Average precision(AP) is defined before Mean Average Precision.

AP is defined as

$$\boldsymbol{AP} = \int_0^1 P(R)dR \qquad \text{Eq: 5.1}$$

mAP is defined as

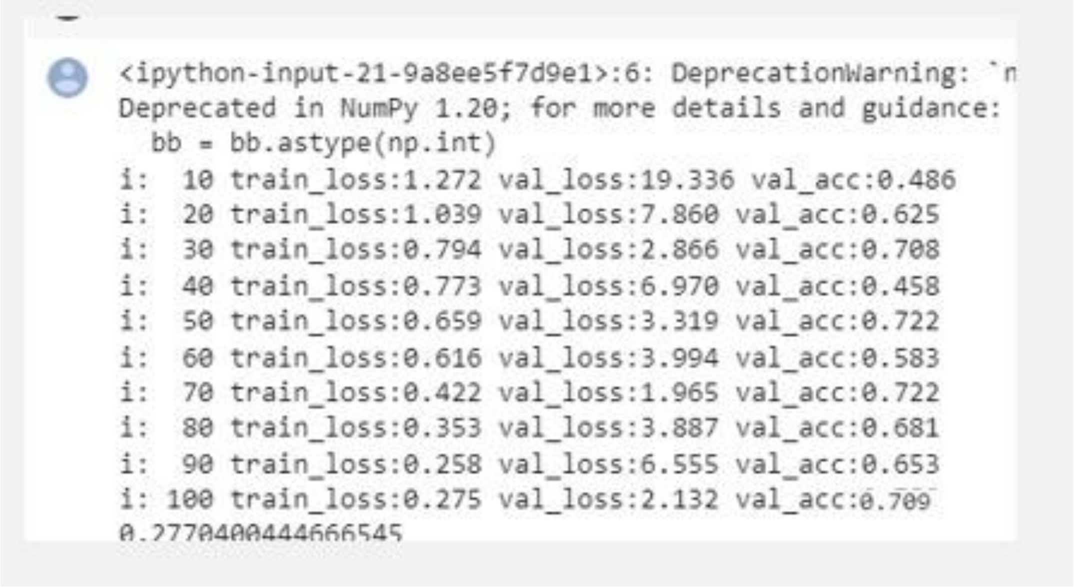
$$\boldsymbol{mAP} = \frac{\boldsymbol{1}}{\boldsymbol{N_{cls}}} \sum_{\boldsymbol{i=1}}^{\boldsymbol{N_{cls}}} AP_i \qquad \text{Eq: 5.2}$$



```
 Epoch    GPU_mem   box_loss   obj_loss   cls_loss  Instances       Size
 96/99      2.07G    0.03271    0.02496   0.009515         45        416: 100% 43/43 [00:06<00:00,  6.35it/s]
            Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 7/7 [00:01<00:00,  5.18it/s]
              all        212        310      0.644      0.724      0.721      0.461

 Epoch    GPU_mem   box_loss   obj_loss   cls_loss  Instances       Size
 97/99      2.07G    0.03264    0.02468    0.01037         44        416: 100% 43/43 [00:06<00:00,  6.20it/s]
            Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 7/7 [00:01<00:00,  5.45it/s]
              all        212        310      0.671        0.7      0.718      0.464

 Epoch    GPU_mem   box_loss   obj_loss   cls_loss  Instances       Size
 98/99      2.07G    0.03055    0.02481   0.009148         29        416: 100% 43/43 [00:06<00:00,  6.27it/s]
            Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 7/7 [00:01<00:00,  5.42it/s]
              all        212        310      0.682      0.692      0.713       0.46

 Epoch    GPU_mem   box_loss   obj_loss   cls_loss  Instances       Size
 99/99      2.07G    0.03207    0.02556     0.0087         44        416: 100% 43/43 [00:06<00:00,  6.37it/s]
            Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 7/7 [00:01<00:00,  5.37it/s]
              all        212        310      0.681      0.695      0.750      0.451
```

Fig 3: Performance of YOLOV5

The Fig 3 shows that the YOLOV5 model has resulted in mean average precision (mAP) of 0.75 that indicates the model is able to detect objects with a high level of accuracy.



```
<ipython-input-21-9a8ee5f7d9e1>:6: DeprecationWarning: `n
Deprecated in NumPy 1.20; for more details and guidance:
  bb = bb.astype(np.int)
i:  10 train_loss:1.272 val_loss:19.336 val_acc:0.486
i:  20 train_loss:1.039 val_loss:7.860 val_acc:0.625
i:  30 train_loss:0.794 val_loss:2.866 val_acc:0.708
i:  40 train_loss:0.773 val_loss:6.970 val_acc:0.458
i:  50 train_loss:0.659 val_loss:3.319 val_acc:0.722
i:  60 train_loss:0.616 val_loss:3.994 val_acc:0.583
i:  70 train_loss:0.422 val_loss:1.965 val_acc:0.722
i:  80 train_loss:0.353 val_loss:3.887 val_acc:0.681
i:  90 train_loss:0.258 val_loss:6.555 val_acc:0.653
i: 100 train_loss:0.275 val_loss:2.132 val_acc:0.709
0.2770400444666545
```

Fig 4: Performance of FasterRCNN

The Fig 4 shows the Faster RCNN result. The model has given an accuracy of 70%. The performance of deep learning models for remote sensing object detection have been analyzed. Two classification algorithms are trained which achieved an accuracy of 83% and 62%. For object detection, YOLOV5 and Faster RCNN were used, which achieved mean average precision (mAP) of 75% and 70%, respectively. The results presented in the tables below clearly indicate the superior performance of YOLOV5 over Faster R-CNN.

TABLE I RESULTS OF RESNET101 AND ZFNET

Model	Training Accuracy	Validation Accuracy
ResNet 101	89%	88%
ZFNet	87%	62%

TABLE II RESULTS OF YOLOV5 AND FASTER RCNN

Model	Mean average Precision(%)
Yolo V5	75
Faster RCNN	70

B. Performance results of YOLOV5 algorithm:

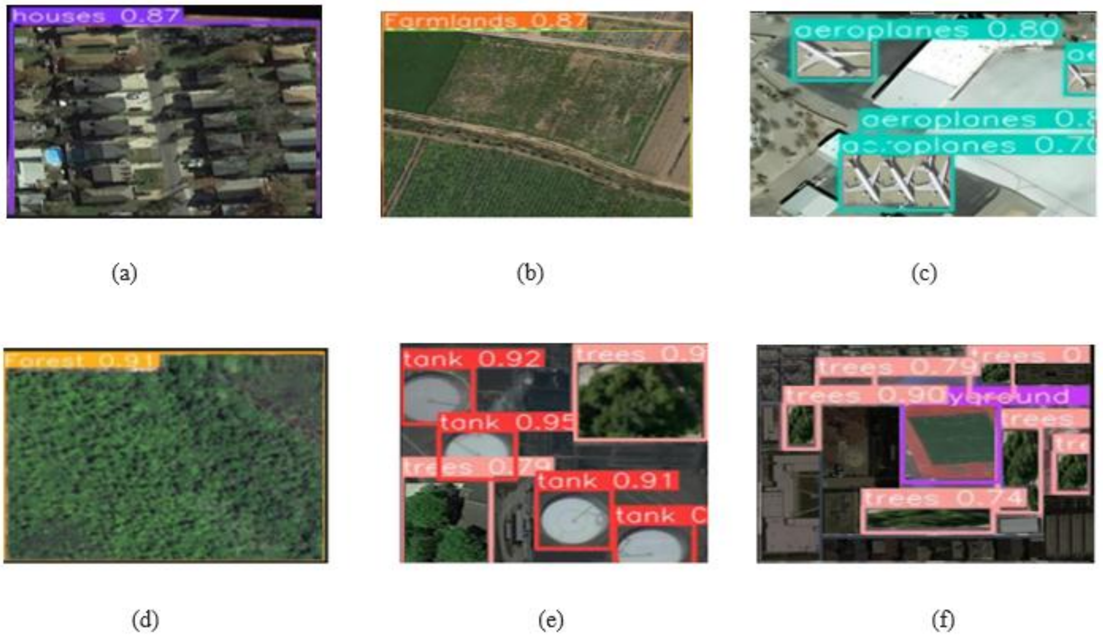


Fig 5: Detection results of YOLOV5 on aerial images dataset. (a) Residential Area. (b)Farmlands (c) Aeroplanes (d) Forest Area (e) Trees and Storage Tanks (f) Playground and Trees.

VI. CONCLUSION

The proposed system uses the customised data for training the model using object detection algorithms such asFaster Regions with Convolutional Neural Networks, You Only Look Once(YOLOV5) and also research the efficiency of deep learning techniques such as ResNet101, ZFNet on Remote Sensing Object Detection. The system also detects objects of different scale variations from aerial images using the above mentioned algorithms. Feature extraction techniques such as ResNet101 has been used which got a higher accuracy of 88%. The Mean Average Precision obtained by YOLOV5 is 75% and for FasterRCNN is 70%. YOLOV5 has given better results when compared to FasterRCNN. The future enhancement of the projectwould be the incorporation of detecting the small objects effectively and transfer learning for the algorithms.

REFERENCES

[1] G. Cheng, J. Han, P. Zhou, and L. Guo, “Multi-class geospatial object detection and geographic image classification based on collection of part detectors, ” ISPRS J. Photogramm. Remote Sens., vol. 98, pp. 119–132, Dec. 2014.

[2] G. Ganci, A. Cappello, G. Bilotta, and C. Del Negro, “How the variety of satellite remote sensing data over volcanoes can assist hazard monitoring efforts: The 2011 eruption of nabro volcano, ” Remote Sens. Environ., vol. 236, Jan. 2020, Art. no. 111426.

[3] J. Ding et al., “Object detection in aerial images: A large-scale benchmark and challenges, ” IEEE Trans. Pattern Anal. Mach. Intell., early access, Oct. 6, 2021, doi: 10.1109/TPAMI.2021.3117983.

[4] J. Redmon and A. Farhadi, “YOLO9000: Better, faster, stronger, ” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 6517–6525.

[5] J. Redmon and A. Farhadi, “YOLOv3: An incremental improvement, ” 2018, arXiv:1804.02767.

[6] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection, ” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 779–788.

[7] K. Li, G. Wan, G. Cheng, L. Meng, and J. Han, “Object detection in optical remote sensing images: A survey and a new benchmark, ” ISPRS J. Photogramm. Remote Sens., vol. 159, pp. 296–307, Jan. 2020.

[8] Q. Wang, J. Gao, and Y. Yuan, “A joint convolutional neural networks and context transfer for street scenes labeling, ” IEEE Trans. Intell. Transp. Syst., vol. 19, no. 5, pp. 1457–1470, May 2018.

[9] Q. Wang, J. Gao, and Y. Yuan, “Embedding structured contour and location prior in siamesed fully convolutional networks for road detection, ” IEEE Trans. Intell. Transp. Syst., vol. 19, no. 1, pp. 230–241, Jan. 2017.

[10] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks, ” IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[11] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, “Focal loss for dense object detection, ” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Oct. 2017, pp. 2999–3007.

[12] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, “Feature pyramid networks for object detection, ” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 2117–2125.

[13] T. Kong, F. Sun, C. Tan, H. Liu, and W. Huang, “Deep feature pyramid reconfiguration for object detection, ” in Proc. Eur. Conf. Comput. Vis. (ECCV), Sep. 2018, pp. 169–185.

[14] W. Xie, J. Lei, S. Fang, Y. Li, X. Jia, and M. Li, “Dual feature extraction network for hyperspectral image analysis, ” Pattern Recognit., vol. 118, Apr. 2021, Art. no. 107992.

[15] W. Xie, J. Lei, Y. Cui, Y. Li, and Q. Du, “Hyperspectral pansharpening with deep priors, ” IEEE Trans. Neural Netw. Learn. Syst., vol. 31, no. 5, pp. 1529–1543, May 2020.

[16] W. Xie, X. Zhang, Y. Li, J. Lei, J. Li, and Q. Du, “Weakly supervised low-rank representation for hyperspectral anomaly detection, ” IEEE Trans. Cybern., vol. 51, no. 8, pp. 3889–3900, Aug. 2021.