Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 10 Issue 06, June-2021

Detection and Classification of Objects in Satellite Images using Custom CNN

Deepthi S.
Scholar, Department of Computer Science & Engineering
Cambridge Institute of Technology,
VTU Bengaluru, India

Prof. Sandeep Kumar
Department of Computer Science & Engineering
Cambridge Institute of Technology,
VTU Bengaluru, India

Dr. Suresh L.
Professor & Head of Dept. of Information Science & Eng.,
RNSIT, VTU Bengaluru,
India

***Abstract*— Satellite image analysis is being increasingly used for many applications like surveillance, military, geo-spatial surveys and environmental impacts and change monitoring. Automatic detection and classification of objects is an important functionality of satellite image analysis. Due to the nature and size of objects and the varied visual features, it becomes challenging to detect and classify objects in aerial images. Manual detection of objects in these images is very time-consuming due to the nature and that data captured in these images. It is desirable to automate the detection of various features or objects from these satellite images. The conventional methods for object classification involve two stages: (i) identify the regions with object presence in the image and (ii) classify the objects in the regions. Additionally, detection of objects becomes challenging in presence of complexities in background, size, noise, and distance parameters. This work proposes a customized convolutional neural network to detect and classify three different objects such as trees, building and cars in the images. It also aims to understand and outline briefly the performance characteristics of the considered custom CNN.**

Keywords*— *Satellite imagery, object detection, classification, custom, convolutional neural networks, image processing

I. INTRODUCTION

Satellite imaging is gaining importance in many applications like remote surveillance, environmental monitoring, aerial survey etc. All these applications involve searching objects, event of interest, facilities etc., from the satellite images. In most applications, manual detection and classification of objects becomes very difficult especially with large volumes of data and the number of satellite images to process collectively. Though detection and classification of objects in images is a long-studied topic in image processing domain, detection in satellite (aerial) images is more challenging as the objects are small and their visual features are extremely hard to track and capture making it all the more difficult. Towards this end various automated techniques for detection and classification have been proposed and are in the works. From classical machine learning (ML) to current deep learning, many solutions have been proposed for object detection and classification in satellite images. Out of these methods machine learning methods for detection and classification are the most researched over the last few decades. These methods involve extraction of various features from the images and classifying them using ML classifiers. Automated Object detection is still a challenge due to variation in the size of the object, orientation, and background of the target object. Conventional machine learning classifiers involving manual selection of features like HOG, Gabor, Hough transform, wavelet coefficients etc., are not able to address the challenges in automated object detection. Hence, there is a need for an efficient approach, and Deep Learning has shown promising results in achieving the objective of detection and classification using CNN. Recently the Deep learning classification methods which have been proposed for automated object detection with high accuracy are able to learn features automatically from the images instead of manual selection of features. Many deep learning models based on convolutional neural network (CNN) are proposed for detection and classification of objects in satellite images. These models involve two steps. In the first step, the regions of presence of object in the image are detected. In the second step, the objects are classified using convolutional neural network.

In this work, a customized convolutional neural network is proposed to detect and classify objects in satellite images. The model is trained to classify three objects of trees, building and cars in the satellite images. The detection and classification performance are compared with actual execution of YOLO V3 algorithm for the same dataset and some standard benchmark data for other algorithms without execution though. YOLOV3 algorithm combines the detection and classification of objects in a single stage instead of two passes as done in conventional CNN models.

II. RELATED WORKS

The existing deep learning models for detection and classification of objects are surveyed in this section.

In [1] the authors have proposed a variation using a CNN to detect objects. For satellite images specifically they put forth the concept of a rotation invariant region based convolutional neural network. In that before classification of the objects begins the step of normalization of feature representation is taking place to achieve and focus on the concept used of a region (rotation invariant). After the same then classification is carried out which is based now on the

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 10 Issue 06, June-2021

fact that for each image there is a higher complexity on computation for patching results through rotation invariant regions for each image.

Authors in [2] are outlining a method for detection of bridges and large crossovers on bodies of water which are shown in satellite images. Water bodies primarily here are rivers which need to be detected or recognized firstly in the image by a technique called as recursive scanning which uses geometric constraints for identifying such details like it is a river type of water body. Thereafter using these identified rivers in the first step, and on the basis of application of the knowledge relate to spatial dimensions concerning different bridges a scan is performed over the extent of the identified rivers to further detect or identify the pixels which could be belonging to a bridge. Once the pixels are identified then an analysis is performed as to the relationship or connectivity of the identified pixels and based upon that analysis it is determined whether a bridge segment is identified in the image for the pixels which got identified first. Although one problem exists for this kind of an approach and that is that we need to have a prior knowledge and understanding of the spatial dimensions of the objects or structures we are trying to identify, which in this case is bridges.

Authors in [3] have detailed a two staged approach for detecting networks of road that are seen or exist in aerial images like ones taken from satellite or UAVs. It leads to automatic detection through first the detection stage and then cutting down or pruning stage being applied. A Bayesian model is used first for classification of shapes of regions which are homogeneous or similar characteristic regions or shapes by detecting these in this stage. The second stage comprises of ascertaining the likelihood of any particular part or segment being a road through the use of a technique called conditional probability. This technique like many others used in detection is very high in its computational complexity.

In [4] the authors propose an image analysis technique which is object based and is used in finding the land cover and its primary usage for classification of the topology of different stretches of areas in the satellite images. Before an SVM classifier can be trained we need to extract texture features and for that all the objects from the image are first segmented. Once this is done, then these segmented objects are used for carrying out the action of extraction. Once the segmented objects' texture features are extracted it is used to train an SVM classifier for the purpose of classifying the land based on its cover and usage. Relevant to this scheme there is a limitation on accuracy due to feature extraction, which is normally constrained and not exhaustive or very complete or accurate in itself.

Authors in [5] have detailed in their work a weakly supervised approach for the same task of detecting objects in satellite images. Boltzmann machine which belongs to the class of generative models with two of the fundamental functions of encoding and decoding is used. The first function of encode takes the features of any image and then these are passed to a Bayesian framework for the purpose of developing the ability or to detect the objects. The Bayesian framework is supposed to be trained with data sets which consist a sampling of images with objects which are classified with different labels for the purpose of training in a mode which is semi supervised.

Authors in [6] have combined a finite state machine with a deep CNN. They have a deep learning network which does the function of extracting from satellite images, road segments. Next, from the image patches classification of road segments is carried through by a trained deep CNN. With the output of this deep CNN and interaction using a finite state machine, combining into set of image batches with the best fit set to outline the road network in the images. Thus, with this combination of finite state machine and deep CNN the proposed technique or method is extracting road segments with a greater degree of accuracy. Nevertheless, this technique too suffers with the same characteristic wherein to achieve accuracy it has to spend an enormous amount of computation power and is complex to find the image patches for the best fit purpose of the object being identified.

Authors [7] proposed a deep learning model to work on aerial images and analyze and completed detailed assessments and localization of damaged structures or buildings. A CNN for this purpose is specifically trained with custom data set which comprises of aerial or satellite images of damaged buildings which are partially or fully fallen, in decrepitude, abandoned or with incomplete structures. Due to the very high computational complexity for satellite images (VHR) due to the amount of data involved, the accuracy also tends to be high.

Authors in [8] proposed a deep learning network which is optimized and attempts at reducing the complexities involved with respect to computation which is a very common occurrence across different classes and hybrid models of CNNs. Here the optimization is specifically for classification of scenes and images in high spatial resolution remote-sensing images which are usually taken by UAVs and satellite sensors and is a topic of immense interest and application currently. The main method to reduce the complexity was by the use of an incrementally efficient convolutional layers with a reduced kernel to minimize or reduce as much as possible the computation complexity. Such an optimized model is claimed to be able to efficiently learn in a robust manner different features for being able to ultimately carry out the task of scene classification. The results that they obtained were promising but more optimization can be performed to reduce the time further from what was obtained for their operation of 40 mins as outlined in the work.

III. EVALUATION OF EXISTING WORK

There are multiple implementations in the field of AI and Deep Learning with which object detection can be performed. But, each of those methods have their own advantages and disadvantages. In this section, the most popularly used implementations such as Faster R-CNN, ResNet and YOLO V3 has been selected. Furthermore, they will be evaluated based on performance metrics namely accuracy, precision, recall and F1 score. In Table 1, the above-mentioned comparisons have been represented.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 10 Issue 06, June-2021

TABLE 1: PERFORMANCE EVALUATION METRICS OF EXISTING SYSTEMS

Implementation	Accuracy	Precision	Recall	F1 score
YOLO V3	90.40	0.90	0.89	0.89
ResNet	87.23	0.87	0.86	0.86
Faster R-CNN	83.64	0.83	0.82	0.82

IV. PROPOSED METHODOLOGY

The architecture of the proposed deep learning model for detection and classification of objects is given in Figure 1. The proposed solution has following important functionalities:

i. Image Acquisition and data processing
ii. Preparation of data and preprocessing
iii. CNN construction and training
iv. CNN validation and analysis of results
v. Drawing of bounding box

The images of building, trees and cars taken from 3D satellite imagery dataset are tagged with labeling tool. The tagged images are used as input for the customized convolutional neural network. The architecture of the customized convolutional neural network and the layer detail of the customized convolutional neural network is given in Figure 2.

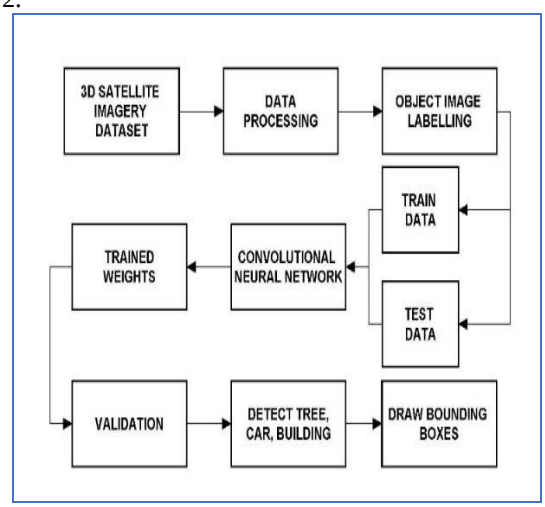

Figure 1 – System Architecture

Since the training needed is resource intensive. The training is done using Google Collab with a Nvidia TESLA T4 Tensor Core GPU with 16GB vRAM. The proposed customized convolutional neural network has the advantage of significant reduction for the same depth in the network for the parameters to be considered, i.e., a reduction in how many parameters for a network with regular convolutions with the same depth.

For a very many numbers of image processing operators, one of the fundamental mathematical operation is Convolution. It is a mathematical operation which is simple but is of a fundamental nature in the realm of image processing. Multiplying together two arrays with equal dimensionality but having different sizes to give a resultant third numbered array which is dimensionally equal can be achieved through the function of convolution. This is often necessary and important aspect which can be used in image processing to be able to realize the implementation of operators which take as input certain pixel values and then give as output pixels which are primarily a simple linear combination of these input pixels. In the context of image processing usually a gray level image is generally represented as an input array with numbers and this forms as just one of the input arrays. A second array called the kernel is normally smaller in magnitude or size and is 2D in terms of dimensionality and could very well be just as thick as a single pixel too.

	Type	Filters	Size	Output
	Convolutional	32	3 × 3	256 × 256
	Convolutional	64	3 × 3 / 2	128 × 128
1×	Convolutional	32	1 × 1	
	Convolutional	64	3 × 3	
	Residual			128 × 128
	Convolutional	128	3 × 3 / 2	64 × 64
2×	Convolutional	64	1 × 1	
	Convolutional	128	3 × 3	
	Residual			64 × 64
	Convolutional	256	3 × 3 / 2	32 × 32
8×	Convolutional	128	1 × 1	
	Convolutional	256	3 × 3	
	Residual			32 × 32
	Convolutional	512	3 × 3 / 2	16 × 16
8×	Convolutional	256	1 × 1	
	Convolutional	512	3 × 3	
	Residual			16 × 16
	Convolutional	1024	3 × 3 / 2	8 × 8
4×	Convolutional	512	1 × 1	
	Convolutional	1024	3 × 3	
	Residual			8 × 8
	Avgpool		Global	
	Connected		1000	
	Softmax			

Figure 2 – Architecture of custom CNN

Convolution is basically the action of sliding this 2x2 or similar kernel across the image from the top left to the other end while ensuring that the kernel is going through all different movements or positions where it does not overflow the boundary of the image, but covers the full image in its movement.

An integral part of CNNs are the pooling layers which help in making the CNNs different and because, as opposed to normal neural networks the CNNs gain the ability of being able to work with and process enormous amounts of data. The fact that features are present in an input image is represented by the convolutional layers. The layers though inherently put forth a problem wherein the location of the features in the input images dictate the problem with respect to the sensitivity of the location of the features. This is the phenomenon which restricts the ability for the method to give good results as it is more attuned to the training data set and gives results which are specific rather than being able to provide data more generically which rather enhances the problem. The problem can be looked at also as overfitting and hence does not provide desired results. That is the reason why the presence of features needs to be generalized over the image and can be done by using the pooling layers as an integral part of the CNN. Both max pooling layer and average pooling serve different purposes wherein the former gives an image which is sharper, driven by focus on the max values, which we can take for this context for example as the intensity of light. While average pooling focuses on giving a smoother image while attempting

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 10 Issue 06, June-2021

to retain the core of the image features. For this specific use-case, avg-pooling has been chosen because it can smoothly extract certain features from a given input image.

System Architecture is a base for building a plan or the design in which the software is created, that can interact with other subsystems to complete the implementation process. It is involved with other subsystems in the work and a framework is designed so that it can communicate and control among them. The main modules that make the system are given in Figure 3.

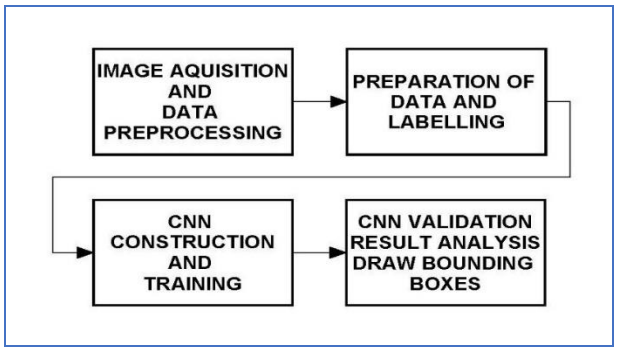

Figure 3 – System Modules

A. Image Acquisition and Data Preprocessing

In the development of any object detection-based system, acquiring the image dataset is vital. While acquiring datasets, some aspects need to be followed that can help for prediction and to train correctly on learning the convolutional neural network model. Generally, more data in the dataset would achieve a better result. The data collected from VALID website is totally un-processed and raw. Furthermore, real-world satellite imagery data is not open for public use, so a 3D modelled version of the same was used in this work to demonstrate the proof of concept.

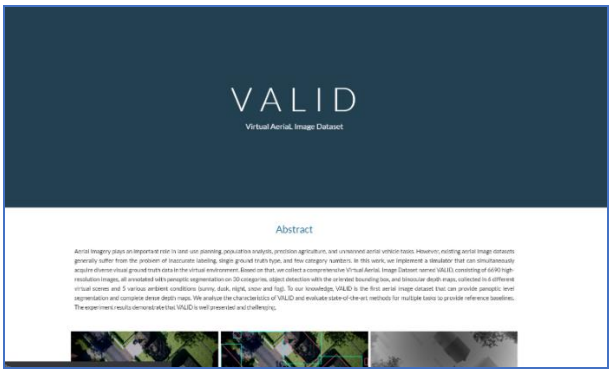

Figure 4 VALID Imagery Dataset

B. Preparation of Data and Object Image Labelling

Preparation of data is defined as the process of acquiring and using the domain knowledge of data to sort and iterate through the images in the dataset. The steps that are involved in processing the raw data are listed and explained in detail in this and further sections below. The huge amount of image data collected from the 3D satellite imagery dataset is then passed through data preprocessing, where we convert this raw data, and clean it to build a dataset that is trainable. Each image is opened and labelled based on the objects present in the image. Here, the labels are stored in a separate XML file, and there exists one such file for every image in the dataset. This XML file contains the coordinates of the objects present in that image. During the training phase, it tells the neural network where exactly the object (such as tree, car, and building) is present.

Labelling of objects in the images is required because the system implementation of this work is based on supervised learning. It is a subcategory of AI and is also well known as supervised deep learning. The method it uses to accurately predict outcomes or for the activity of classification of data is primarily by using labeled datasets in the training of algorithms. The algorithm is a semi self-learning algorithm or model wherein it keeps adjusting the weights based on the input data given to it till the time that the model is seen or taken to be fitted appropriately. This process of learning and adjusting occurs during the process of cross validation. Organizations can solve a variety of problems pertaining to the real world both of complexity and scale, with the help of supervised learning, for example like identifying and classification of different types of mails giving priority or marking as spam or other marketing or phishing kind of mails in one's email inbox.

During the training phase, it tells the neural network where exactly the object (such as tree, car, and building) is present. Labelling of objects in the images is required because the system implementation of this work is based on.

C. CNN Construction and Training

The Training phase is the subset of the training a model and the Test set is the subset to test on the trained model. CNN construction is the process of custom designing a neural network architecture that can facilitate this particular use-case. In this way we can give the CNN the training data as the input which should contain the correct answer, known as the target attribute. This work takes advantage of a custom neural network designed for satellite imagery. In Fig 6, implementation of the custom CNN has been implemented in code.

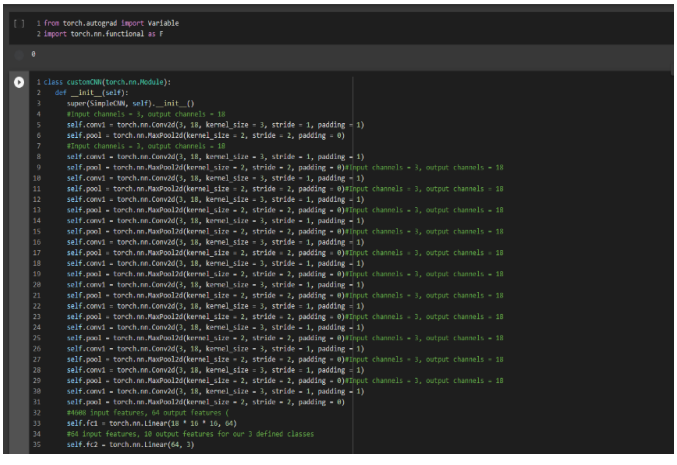

Figure 5 Code Implementation of CNN Layers

Training was done with about 90% of the 3D Satellite Imagery Dataset, and the rest 10% was used for testing purpose. This gives the CNN more amount of data to train upon, which leads to good prediction accuracy. This predicted value's weighted average is calculated iteratively to get the final predicted value. As this specific system needs to

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 10 Issue 06, June-2021

detect objects such as cars, buildings and trees, it comes under the custom object detection sub-class. Here, training is really resource intensive and could not be done on a local machine or laptop. So, to tackle this problem a VM was used within Google Collab with a Nvidia TESLA T4 Tensor Core GPU with 16GB vRAM. Training was done two different times, with two different epochs for each training iteration, which were 50 and 100 epochs respectively. In Fig 7, the screenshot shows the training done with about 100 epochs. After each epoch, the net number of objects trained in that epoch will be listed.

Upon successful completion of training, the custom CNN gives a (.pt) weights file, which can be used with the help of inference method to perform detection on random object. To change input data from within the multiple layers which are hidden in the network a parameter such as weight is very important in the working of a neural network In other words the network is nothing but interconnected neurons like in one's brain or a collection of nodes, carrying and processing information or data. Each node is characterized by three factors which are there within namely weight, a set of inputs, and a bias value. In a sequential processing like an assembly line process, a node processes the input where the operation of multiplication with the weight is carried out resulting into an output. Then a decision-making process is embarked upon to understand if the output should be experimented with or passed to the next layer in the CNN. The weights are often contained in the layers of the network which are hidden. So, the weights are set in such a way that the system can effectively detect the listed objects (cars, trees, buildings) from the satellite images given as an input.

D. CNN Validation, Analysis of Results

One of the most important phases of testing the model to the input set of data, using validation. After the training process, the test data is used to compare the ground truth with the predicted outcome, from the CNN's predicted output prediction.

Bounding box's function can be implemented with raw python code and is deployed using a function called as "visualize_detections". This specific function takes in a few parameters as in input such as the images, boxes, the classes, scores for each prediction, the figure-size, linewidth and color of the bounding boxes as RGB (Red-Green-Blue) channels. These results are basically metrics that can facilitate with the analysis of prediction results. Results also print the names of the objects that it was able to successfully detect. In this work the result is the detect of the objects (cars, trees, buildings) and how accurately the system is able to predict these objects.

V. RESULTS

In this stage, the proposed system can be tested with a few random images from the test folder files given to it as an input, and let it identify and detect the objects such as car, trees, and buildings in it, and draw bounding boxes around them to make it easy for identification. For detection, Google Collab or CLI is used with Python virtual environment.

The object detection phase of the system is shown with 50 epochs and similarly also with 100 epochs. We can see the system is trained with 100 epochs in total, which yielded a far better accuracy, which is also the final prediction accuracy obtained. Metrics from the Tensor-Board library were used for the evaluation of the model with the last trained weights with the highest accuracy and precision was given out as an output and the final accuracy achieved is 94.65%. The highest peak will be considered here.

Figure 6 System during run-time (with 50 epochs)

Figure 7 System during run-time (with 100 epochs)

In Figure 8, the final evaluation metrics have been represented with the accuracy of 94.65%.

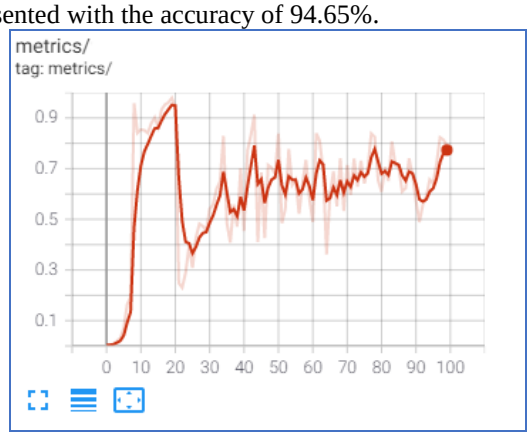

Figure 8 Accuracy of system with 100 epochs

The precision plot over different confidence in the proposed custom CNN for different class of object is given below.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 10 Issue 06, June-2021

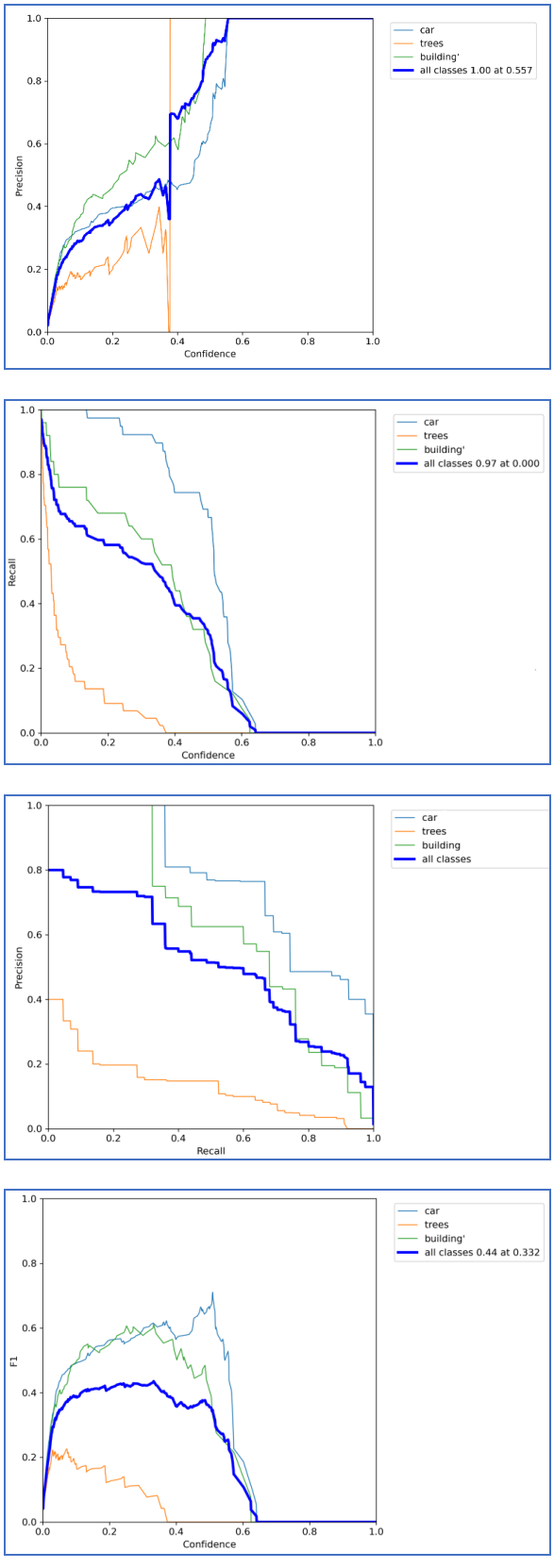

Figure 9 - P vs. R Graphs

A better balance between precision and recall is achieved for car object followed by building and trees in the proposed solution:

- **Precision plot over different confidence** - Car objects are classified with higher confidence in the proposed custom CNN, followed by buildings and trees.
- **Ration between precision and recall** - The recall is higher for car followed by building and tress objects in the proposed custom CNN solution.
- **F1 measure plot for different class of objects** - From the results, car and building objects are classified far better compared to trees in the proposed solution.

After successful detection, a comparison graph was plotted to compare the proposed custom convolutional neural network-based implementation to the other implementation mentioned in the comparison, in Figure 10 below. From comparison, we can clearly observe the edge the custom CNN has over the other implementations for detection of objects from satellite imagery. Furthermore, YOLO v3 and 4 were tested extensively for the same use-case scenario, but it yielded just around 90% net prediction accuracy. The proposed custom CNN solution has 4.25% more accuracy compared to YOLO v3.

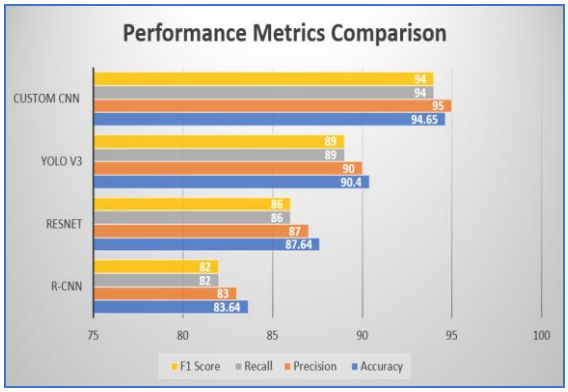

Figure 10 – Comparative Analysis of Performance Metrics

VI. CONCLUSION

This work proposed a custom CNN model for detection and classification of objects in satellite images. The performance of the proposed solution was tested for three different objects of car, building and tress. The method was able to achieve an accuracy of 94.65%. The volume of training image used in this work is small, and as future work we plan to test the performance of the model against large volume of datasets. With respect to the future development that can be applied to this work to take it to the next level, real-time satellite imagery can be used (with the required permissions sanctioned from the respective space agency) to train the proposed neural network. This way, the proof-of-concept system will be ready for production deployment.

ACKNOWLEDGMENT

I wish to extend my thanks to Prof. Sandeep Kumar, Dept. of CSE, Cambridge Institute of Technology (CIT) for his guidance and impressive technical suggestions, Dr. Suresh and all my professors and colleagues at Dept. of CSE, CIT in supporting my project and in publication of this paper. I also wish to thank my husband for supporting me during this time.

REFERENCES.

[1] Cheng, G.; Zhou, P.; Han, J. Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images. IEEE Trans. Geosci. Remote Sens. 2016.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 10 Issue 06, June-2021

[2] Chaudhuri, D., Samal, A., An automatic bridge detection technique for multispectral images. IEEE Trans. Geosci. Remote Sens. 2008
[3] Hu, J., Razdan, A., Femiani, J.C., Cui, M., Wonka, P., Road network extraction and intersection detection from aerial images by tracking road footprints. IEEE Trans. Geosci. Remote Sens., 2007
[4] Goodin, D.G., Anibas, K.L., Bezymennyi, M., 2015. Mapping land cover and land use from object-based classification: an example from a complex agricultural landscape. Int. J. Remote Sens. 36, 4702-4723.
[5] Han, J., Zhang, D., Cheng, G., Guo, L., Ren, J., Object detection in optical remote sensing images based on weakly supervised learning and high-level feature learning. IEEE Trans. Geosci. Remote Sens. 2015.
[6] Wang, J., Song, J., Chen, M., Yang, Z., Road network extraction: a neural-dynamic framework based on deep learning and a finite state machine. Int. J. Remote Sens.2015
[7] Duarte, D., et al. "Satellite image classification of building damages using airborne and satellite image samples in a deep learning approach." ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences 4.2 (2018).
[8] Zhong, Y.; Fei, F.; Liu, Y.; Zhao, B.; Jiao, H.; Zhang, L. SatCNN: Satellite image dataset classification using agile convolutional neural networks. Remote Sens. Lett. 2016, 2, 136–145.
[9] Arshitha Femin and Biju K.S., "Accurate Detection of Buildings from Satellite Images using CNN" IEEE, 2020.
[10] G. Scott, M. England, W. Starms, R. Marcum, and C. Davis, "Training deep convolutional neural networks for land–cover classification of high-resolution imagery", IEEE Geoscience and Remote Sensing Letters, vol. 14, no. 4, 2017.
[11] F. Hu, G.-S. Xia, J. Hu, and L. Zhang, "Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery," Remote Sens., vol. 7, no. 11, pp. 14 680–14 707, 2015.
[12] F. Luus, B. Salmon, F. van den Bergh, and B. Maharaj, "Multiview deep learning for land-use classification," IEEE Geosci. Remote Sens. Lett., vol. 12, no. 12, pp. 2448–2452, Dec. 2015
[13] Kadhim, Mohammed & Abed, Mohammed, Convolutional Neural Network for Satellite Image Classification, 2020.
[14] Imamoğlu, Nevrez & Martinez-Gomez, Pascual & Hamaguchi, Ryuhei & Sakurada, Ken & Nakamura, Ryosuke. (2018). Exploring Recurrent and Feedback CNNs for Multi-Spectral Satellite Image Classification. Procedia Computer Science. 140. 162-169. 10.1016/j.procs.2018.10.325.
[15] Liang, Ming and Xiaolin Hu (2015) "Recurrent Convolutional Neural Network for Object Recognition" IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'15).
[16] Tan, Y.; Xiong, S.; Li, Y. Automatic Extraction of Built-Up Areas from Panchromatic and Multispectral Remote Sensing Images Using Double-Stream Deep Convolutional Neural Networks. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 2018, 11, 3988–4004
[17] Li, Y.; Zhang, Y.; Huang, X.; Yuille, A.L. Deep networks under scene-level supervision for multi-class geospatial object detection from remote sensing images. ISPRS J. Photogramm. Remote Sens. 2018
[18] L. Mou, P. Ghamisi and X. X. Zhu, "Deep Recurrent Neural Networks for Hyperspectral Image Classification," in IEEE Transactions on Geoscience and Remote Sensing, vol. 55, no. 7, July 2017
[19] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016
[20] Bochkovskiy, Alexey & Wang, Chien-Yao & Liao, Hong-yuan. (2020). YOLOv4: Optimal Speed and Accuracy of Object Detection., 2020
[21] Chien-Yao Wang, Hong-Yuan Mark Liao, Yueh-Hua Wu, Ping-Yang Chen, Jun-Wei Hsieh, and I-Hau Yeh. CSPNet: A new backbone that can enhance learning capability of cnn. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshop (CVPR Workshop), 2020.