iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 12 **Issue:** VI **Month of publication:** June 2024

DOI: https://doi.org/10.22214/ijraset.2024.63096

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue VI June 2024- Available at www.ijraset.com

Satellite Image Classification using TensorFlow

Darshan P[1], Dinesh[2], Akhil Prasad[3], Bhargav S[4], Dr. Asha[5]
Dr. Ambedkar Institute of Technology, Bangalore-560072, Karnataka, India

Abstract: ***Satellite image classification plays a crucial role in various fields such as agriculture, urban planning, disaster management, and environmental monitoring. This paper presents a novel approach utilizing TensorFlow, a popular open-source machine learning framework, for satellite image classification. The proposed methodology leverages deep learning techniques to extract meaningful features from satellite images, enabling accurate classification into predefined categories. By harnessing the power of convolutional neural networks (CNNs) implemented in TensorFlow, this research aims to enhance the efficiency and accuracy of satellite image classification tasks. The experimental results demonstrate the effectiveness of the proposed approach in achieving high classification accuracy of about 96.5% while maintaining computational efficiency. It classifies about 10 different classes obtained from the EuroSat dataset. It is implemented in Google Collab for faster training and implementation.***
Keywords: Artificial Intelligence (AI); convolutional neural network (CNN); machine learning; classification; satellite imagery

I. INTRODUCTION

Satellite image classification is a fundamental task in remote sensing that involves assigning semantic labels to different regions of an image. With the increasing availability of high-resolution satellite imagery and the rapid progress in deep learning, there has been growing interest in developing advanced techniques for satellite image classification. Deep learning is a family of machine learning models that use many processing layers to represent input at various degrees of abstraction [1].By fusing potent graphical processing units (GPUs) with massive neural network models, known as convolutional neural networks (CNNs), it has accomplished astounding success in object detection and categorization. The yearly ImageNet Large Scale Visual Recognition Challenge has been dominated by CNN-based algorithms for object detection and classification in photos since 2012 [2]. As seen in Fig. 1, a CNN is made up of several processing layers. A set of convolution filters that identify features in an image make up each layer. Higher-level feature detectors are formed by subsequent layers, which resemble the Gabor-like and color blob filters seen in Fig. 2 in the early levels. Towards the end of the series, the CNN creates a set of predicted probabilities, one for each class, by combining the detector outputs in fully linked "dense" layers. With CNNs, feature detectors do not need to be engineered by the algorithm creator, in contrast to more traditional techniques like SIFT [3] and HOG [4]. As the network trains, it learns which features to look for and how to look for them. Advanced GPUs supply the processing power needed for such huge CNNs. Fast GPUs and open-source deep learning software libraries like TensorFlow [5] and Keras [6] have contributed to the field's ongoing progress.

A. Problem

Finding interesting objects, buildings, and events in satellite photography is a significant issue. Law enforcement organizations look for unauthorized mining operations or illegal fishing vessels; disaster response teams want to map flooding or mudslides; and financial investors want more effective ways to keep an eye on the drilling of oil wells or development of agricultural land. Automation is necessary because there are many geographical areas to cover and not enough analysts to perform the searches. However, the issue cannot be resolved by using conventional object identification and classification methods because they are too erroneous and unstable. Deep learning has had only patchy success when applied to satellite imagery, despite several recent attempts. The challenge of efficiently preparing satellite imagery for CNN input is one of the causes.

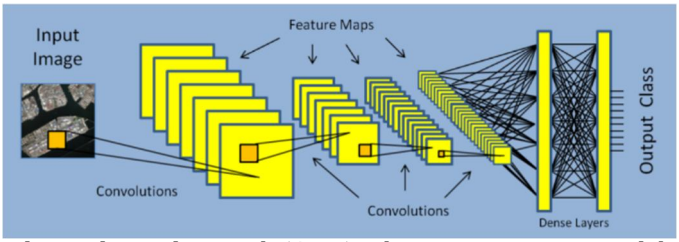


Fig. 1. The structure of a convolutional neural network (CNN). The input image is passed through a series of image feature detectors.

©IJRASET: All Rights are Reserved

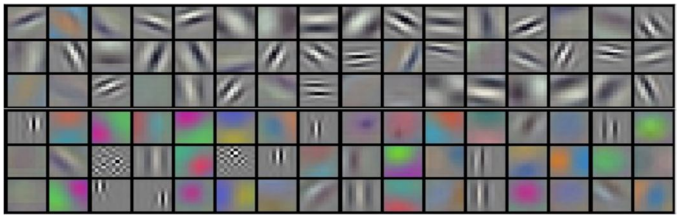


Fig. 2. Examples of the image feature detectors that a CNN might "learn" during its training [6].

II. PRIOR WORKS

Deep learning has had only patchy success when applied to satellite imagery, despite several recent attempts. The challenge of efficiently preparing satellite imagery for CNN input is one of the causes. For example, the EuroSat dataset comprises 27000 aerial images based on Sentinal-2 satellite images. The 10 different classes include annual and peranent crop, froest, herbaceous vegetation, highway, industrial, pasture, residential, river, and sea lake. The images are 64x64 pixels in size and has 3 color channels (typically red, gree, and blue, often represented as RGB).

For instance, 2100 aerial photos from the US Geological Survey are included in the UC Merced Land Use Dataset [7, 8]. The photos have a pixel resolution of 256x256 and a ground sample distance of 0.3 meters. The 21 classes include facility classes like storage tanks and tennis courts as well as land cover kinds like road, water, and farmland. The UC Merced photos were classified into land cover types by a number of researchers using the VGG, ResNet, and Inception CNNs [9–11]. One researcher obtained classification accuracies as high as 98.5% [11]. However the size, number and types of classes, and geographic variety of this dataset are quite constrained. Building footprints and high-resolution DigitalGlobe satellite photos of five cities make up the SpaceNet dataset [12]. CNNs have been trained to identify building footprints by segmenting the images [13]. The geographic scope and usefulness of this dataset for classifier training are both constrained. A list of further remote sensing datasets can be found in Ref. [14]. Not one of them has the hundreds of thousands of photos worldwide needed to create a flexible image classification system.

III. DATASET

In addition to NASA and its Landsat mission, the European Space Agency (ESA) is stepping up its Copernicus initiative to enhance Earth observation. ESA runs a fleet of satellites known as Sentinels as part of this mission.

In this study, we address the problem of classifying land use and land cover using RGB picture data from the Sentinel-2A satellite. Sentinel-2A is one of two similar land monitoring satellites in the constellation Sentinel-2A and Sentinel-2B. June 2015 saw the successful launch of Sentinel 2A, while March 2017 saw the launch of Sentinel-2B. There are two variants of Sentinal-2 dataset, one is rgb variant and the other is 13 spectral band variant. We will be using the rgb variant dataset for faster classification process.

IV. METHODS

A. EfficientNetV2 Architecture Design

The architecture for the model we searched, EfficientNetV2-S, is displayed in Table below. Our searched EfficientNetV2 has a few key differences from the EfficientNet backbone: (1) The first distinction is that in the early layers, EfficientNetV2 heavily utilizes fused-MBConv (Gupta & Tan, 2019) in addition to MBConv (Sandler et al., 2018; Tan & Le, 2019a). (2) Secondly, since smaller expansion ratios often have lower memory access overhead, EfficientNetV2 favors smaller expansion ratios for MBConv. (3) Thirdly, EfficientNetV2 favors smaller 3x3 kernel sizes; however, in order to offset the decreased receptive field caused by the lower kernel size, it adds extra layers. (4) Finally, the final stride-1 stage in the original EfficientNet is entirely eliminated in EfficientNetV2, possibly as a result of its high parameter size and memory access overhead.

Stage	Operator	Stride	#Channels	#Layers
0	Conv3x3	2	24	1
1	Fused-MBConv1, k3x3	1	24	2
2	Fused-MBConv4, k3x3	2	48	4
3	Fused-MBConv4, k3x3	2	64	4
4	MBConv4, k3x3, SE0.25	2	128	6
5	MBConv6, k3x3, SE0.25	1	160	9
6	MBConv6, k3x3, SE0.25	2	256	15
7	Conv1x1 & Pooling & FC	-	1280	1

Table 1 EfficientNetV2-S architecture – MBConv and Fused MBConv blocks are described in the below figure 3.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue VI June 2024- Available at www.ijraset.com

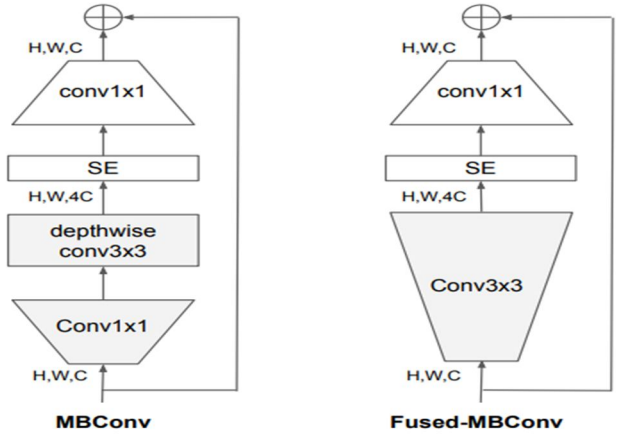


Fig. 3 Structure of MBConv and Fused-MBConv.

This model helps us in training the dataset at faster rate and also in prediction of the input images with higher accuracy making it most efficient.

B. Image Preparation

1) Data Preprocessing

a) Resizing and Normalization: Images are resized to a fixed size and pixel values are normalized to a specific range to ensure uniformity in the dataset.

b) Data Augmentation: Techniques like random cropping, flipping, and rotation are applied to increase the diversity of the training data, which can improve the model's generalization ability.

2) Supervised Classification

a) Training Set Creation: Supervised classification methods require input from an analyst in the form of a training set. The quality and representativeness of the training samples significantly impact the accuracy of the classification methods.

C. Training And Validation

1) Training Process

Dataset Splitting: The dataset is typically divided into training, validation, and test sets. The training set (usually around 60% of the data) is used to train the model, while the validation set (around 20%) is used to tune hyperparameters and prevent overfitting.

Model Training: The pre-processed data is fed into the CNN model, which is initialized with pre-trained weights (e.g., VGG-16 or ResNet or EfficientNetV2). The model is trained using backpropagation and optimization algorithms like Adam with a suitable learning rate and batch size.

Loss Calculation: During training, the model calculates the loss function, which measures the difference between predicted and actual labels. The goal is to minimize this loss by adjusting the model's weights through optimization.

Monitoring Performance: The training process involves monitoring metrics such as loss and accuracy on the training set. This helps in assessing the model's progress and identifying potential issues like overfitting.

2) Validation Process

Hyperparameter Tuning: The validation set is used to fine-tune hyperparameters like learning rate, batch size, and regularization techniques. This process helps optimize the model's performance and generalization ability.

Model Evaluation: The trained model is evaluated on the validation set to assess its performance on unseen data. Metrics like accuracy, F1-score, and confusion matrix are computed to gauge the model's effectiveness in classifying satellite images.

Early Stopping: Early stopping is a technique used during validation to prevent overfitting. If the model's performance on the validation set starts to degrade, training can be stopped early to avoid learning noise in the data.

Model Selection: Based on the validation results, the best-performing model is selected for further evaluation on the test set. This ensures that the model generalizes well to new, unseen satellite images.

©IJRASET: All Rights are Reserved

D. Testing

a) EfficientNetV2 Model Inference

The trained model is used to make predictions on the test dataset (rest 20% of the dataset). Each satellite image is passed through the model, and the model predicts the class label for each image based on the learned features.

b) Performance Evaluation

Various evaluation metrics are calculated to assess the model's performance on the test set. Common metrics include:
Overall Accuracy: The percentage of correctly classified images.
F1-Score: A measure of a model's accuracy that considers both precision and recall.
Confusion Matrix: A table that summarizes the model's performance by comparing predicted and actual class labels.

c) Error Analysis

Errors made by the model during testing are analyzed to identify patterns or specific classes where the model struggles. This analysis can provide insights into areas for model improvement or dataset refinement.

V. RESULT AND DISCUSSION

Once the dataset are imported along with the necessary libraries like TensorFlow, PIL, etc. the dataset is divided using the python function, splice to split the dataset into 60%, 20%, 20%, respectively for training, validating, and testing. The EfficientNetV2 is the pre-trained model which is imported and stored in the model(s) variable, which is then trained for 5 epochs at first.
Now the model is trained the validation begins and acquires the accuracy of about 96.5% and above. The testing phase is carried out to check with the accuracy. The trained model is stored as an h5 file and is used for classification of new inputs and get the results. Currently the model is trained for the dataset containing the images of only RGB values. We can take up the second variant of the EuroSat dataset available i.e., 13 spectral bands and train the model which achieves the accuracy of about 98.3% or higher based on the system specification.
The fig. 4 below shows the sample output image for the given set of inputs in a single snapshot.
Currently the program is set to take up the maximum input of about 32 images at a single time and classify them based on the frequencies of the R,G,B values stored in them, and prints out the class name each images belong to, right above the respective images.
Although the model can still be rendered with by trying out the different model like RCNN, ResNet and many other state-of-the-art model which can accurately classify the images acquired from satellite but can be time consuming making the proposed approach one of the best and faster one.



Fig. 4 Sample output snapshot with the image classifications respectively, for the give set of input images

VI. CONCLUSION

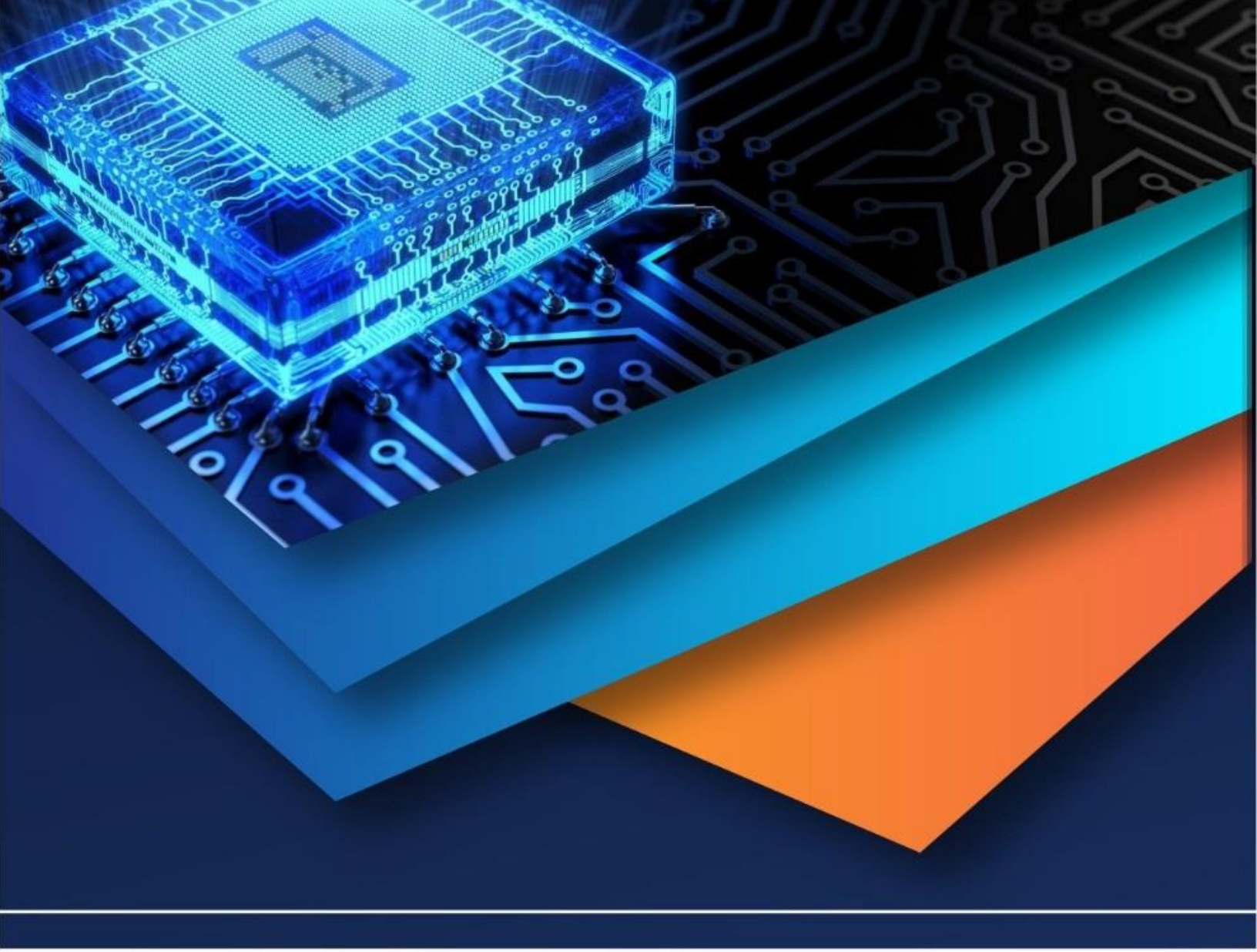
We have presented a deep learning system that classifies land covers and facilities in high-resolution multi-spectral satellite imagery. The system consists of an ensemble of CNNs with post-processing neural networks that combine the predictions from the CNNs with satellite metadata. On the EuroSat Sentinal-2 dataset of 27000 images in 10 classes, including the false detection class, the system achieves an accuracy of 0.966 and an F1 score of 0.96.7. It classifies 10 classes with an accuracy of 95% or better and beats the ResNet model in the fMoW TopCoder challenge.

Combined with a detection component, our system could search large amounts of satellite imagery for land regions. In this way it could solve the problems posed at the beginning of this paper. By monitoring a store of satellite imagery, it could help law enforcement officers detect unlicensed mining operations or illegal fishing vessels, assist natural disaster response teams with the mapping of mud slides or hurricane damage, and enable investors to monitor crop growth or oil well development more effectively.

REFERENCES

[1] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” Nature, vol. 521, pp. 436-444, 28 May 2015.
[2] "ImageNet: The Large Scale Visual Recognition Challenge (ILSVRC)," www.image-net.org/challenges/LSVRC is the URL for ImageNet.
[3] D.G. Lowe, “Distinctive image features from scale-invariant keypoints,” Int. Journal of Computer Vision, vol. 60, no. 2, pp. 91-110, 2004.
[4] D. Navneet and B. Triggs, “Histograms of oriented gradients for human detection,” IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), pp. 886–893, 2005.
[5] “TensorFlow: An open-source software library for machin intelligence,” TensorFlow, https://www.tensorflow.org/
[6] F. Chollet at al., “Keras”, GitHub, 2017, https://github.com/fchollet/ keras.
[7] University of California, Merced, "UC Merced Land Use Dataset," http://weegee.vision.ucmerced.edu/datasets/landuse.html.
[8] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," Proc. The Eighteenth ACM SIGGRATUTAL International Symposium on Geospatial Advances. Info. Sys., pp. 270-279, 3-5 Nov 2010.
[9] Y. Liang, S. Monteiro, and E. Saber, IEEE Workshop on Applied Imagery Pattern Recognition (AIPR), October 2016, "Transfer learning for high-resolution aerial image classification."
[10] M. Castelluccio, G. Poggi, and L. Verdoliva, "Convolutional Neural Networks for Land Use Classification in Remote Sensing Images," arXiv 1508.00092, August 2015.
[11] G. Scott, M. England, W. Starms, R. Marcum, and C. Davis, "Convolutional Neural Networks for Land Use Classification in Remote Sensing Images," arXiv 1508.00092, August 2015, vol. 14, no. 4, pp. 549-553, Apr 2017.
[12] “SpaceNet on AWS,” Amazon.com, https://aws.amazon.com/publicdatasets/spacenet/.
[13] E. Chartock, W. LaRow, and V. Singh, “Extraction of Building Footprints from Satellite Imagery,” Stanford University Report, 2017.
[14] G. Cheng, J. Han, and X. Lu, "State of the Art and Benchmarks for Remote Sensing Image Scene Classification," Proc. IEEE, vol. 105, no. 10, pp. 1865-1883, October 2017.
[15] “Satellite Image Classification with Deep Learning” - Mark Pritt and Gary Chern - Lockheed Martin Space. https://arxiv.org/pdf/2010.06497

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)