P Rakesh Vardhan, 2024, 12:3
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology

An Open Access Journal

Enhancing Satellite Image Classification with CNN

P Rakesh Vardhan, G Srujan Sharma, T Shiva Shankar,
Associate Professor Mr. P. Raveendra Babu
Dept. of ECE,
CMR College of Engineering & Technology, Telangana

Abstract- There exist several applications for satellite images, such as environmental monitoring, law enforcement, and disaster response. For these applications, manual identification within such imagery is essential. Automation is necessary, though, because of the large geographic areas and scarcity of human resources for analysis. For such jobs, traditional object recognition algorithms frequently suffer from poor accuracy and dependability. CNN in particular, are part of deep learning and have shown promise in automating image interpretation tasks. In this study,. In particular, we offer a deep learning method that is intended to categorize dataset items and facilities into 60 distinct types. Our system comprises an ensemble of CNNs augmented Using additional neural networks that combine visual characteristics with satellite information. Python is used for implementation, and the deep learning libraries Keras and Tensor Flow are utilised. attaining an F1 score of 0.797 and an overall accuracy of 83%.

Keywords- Artificial intelligence; AI; advanced neural networks; machine-based learning; image comprehension; identification; categorization; remote sensing images; image processing; computer vision; pattern recognition

I. INTRODUCTION

Deep learning has become a formidable machine learning approach [1]. Especially with the combination of high-performance graphical processing units (GPUs) and convolutional neural networks (CNNs), this paradigm has shown impressive effectiveness in object detection and classification tasks [2]. The Image Net Large Scale Visual Recognition Challenge has been dominated by CNN-based algorithms since 2012, demonstrating their efficacy in identifying and classifying objects in images [3]. Due to this achievement, image understanding has undergone a revolutionary change, and leading tech companies [1].

A CNN architecture is made up of a series of processing layers, each of which has convolution filters built into it that are intended to identify characteristics in images [4]. Class-specific probabilities are produced by these filters, which gradually extract characteristics of ever greater complexity, culminating in completely linked "dense" layers. Notably, CNNs eliminate the requirement for human feature engineering since, during training, the network automatically learns to recognise and characterise features, in contrast to conventional techniques like Scale-Invariant Feature Transform (SIFT) and Histogram of Oriented Gradients (HOG) [5][6]..

The evolution of CNN architectures has seen a trend towards increasing complexity, with notable models such as VGG, Google's Inception, Res Net, and Dense Net boasting unprecedented numbers of layers [7][8][9]]. However, the deployment of such large-scale CNNs is facilitated by advanced GPUs. Open-source deep learning frameworks like Tensor Flow and Keras, coupled with GPU acceleration, have been instrumental in propelling advancements in deep learning.

Applications for satellite images are numerous and include environmental monitoring, law enforcement, and disaster response. However, because of the large geographic coverage and scarce human resources, manually identifying the items and facilities inside these photos is labor-intensive and unfeasible. As a result, deep learning—especially CNNs—offers a viable way to automate these kinds of jobs. In this work, we use CNN-based methods to tackle the problem of multi-spectral, high-resolution satellite images recognition. In particular, we describe a (fMoW) dataset items and facilities into 60 distinct groups. developed in Python with the aid of the Tensor Flow and Keras libraries

II. LITERATURE SURVEY

This paper presents a technique for obtaining unique perspectives on items or scenes. These characteristics are independent of picture rotation and scale, allowing for strong matching under different affine distortions, 3D perspective shifts, noise addition, and lighting variations. The paper also investigates the use of these characteristics for object recognition, using Hough transformations to identify clusters that correspond to a single item after individual feature matching with rapid nearest-neighbor algorithms. Then, least-squares solutions for consistent posture parameters are used for verification. This method performs almost in real-time and shows strong object recognition in the face of occlusion and clutter. Using linear SVMs as a test case, this study looks at feature sets for reliable visual object recognition on human detection. In exploring how each computational step affects performance, the study emphasizes the importance of high-quality local contrast normalisation, fine-scale gradients, somewhat coarse spatial binning, and fine orientation binning in overlapping descriptor blocks. This work trains a deep convolutional neural network on a massive dataset of 1.3 million high-resolution images into one thousand unique classes. The study beats previous state-of-the-art findings with top-1 and top-5 error rates on the test data of 39.7% and 18.9%, respectively. The neural network design consists of five convolutional layers, many max-pooling layers thereafter, two globally connected layers, and a final 1000-way softmax. Training is accelerated by using efficient GPU convolutional net implementations and non-saturating neurons. Furthermore, a new regularisation method is demonstrated that successfully lowers overfitting in the globally connected layers.

The application of DCNNs to the categorization of land cover in remote data is verified in this work. This challenge is addressed by using transfer learning (TL) with data augmentation and fine-tuning for remote sensing imaging, as there are limited remote-sensing picture datasets. Whereas data augmentation makes use of diverse features of remote sensing images to increase training datasets and improve DCNN resilience, training layer augmentation allows bootstrapping of a DCNN while maintaining deep visual feature extraction from a picture corpus in a separate domain. CaffeNet, GoogLeNet, and ResNet land-cover classification accuracies are demonstrated through experimental findings on the UC Merced dataset.

The FCNN generates signed distance labels indicating pixel inclusion/exclusion from building footprints, which are subsequently post-processed to produce bounding polygons. The approach outperforms previous methods, achieving an F1 score of 0.34 on a similar dataset, surpassing the winning implementation of the first SpaceNet Challenge that did not utilize deep learning.

The NWPU-RESISC45 dataset, a large-scale benchmark with 31,500 pictures in 45 scene classes, is reviewed in-depth in this study. The dataset overcomes the shortcomings of previous datasets, such as their lack of variety, small-scale scene classifications, and restricted image counts.

The challenge involves handling various complications such as clutter, occlusion, and diverse scene contexts. Winning solutions employ innovative approaches to robustly identify objects amidst these challenges. This study explores functional representations for shape maps, allowing operations beyond point-to-point mappings. By posing descriptor-based functional constraints, the study achieves near-isometric mappings, even in the absence of landmark constraints. The representation facilitates algebraic operations on shape maps, enabling operations such as subtraction to retrieve orientation-preserving near-isometries.

III. METHODOLOGY

1. Johns Hopkins APL model

In recent years, various annotated datasets of imagery have combined, with efforts made in the areas of detection and classification. When used to remotely sensed data, deep learning techniques have mostly concentrated on classifying land cover or identifying buildings. For instance, the U.S. Geological Survey provided 2100 aerial photographs for the UC Merced Land Use Dataset. These 256 × 256 images' pixel size indicates a 0.3 metre ground sample distance. Researchers have classified UC Merced pictures into 21 classifications of land cover, including as farming, roads, water, and infrastructure types like tennis courts and storage tanks, using VGG, Res Net, and Inception CNNs. There have been reports of classification accuracy as high as 98.5%. However, this dataset has drawbacks due to its small size, lack of class variety, and limited geographic coverage. The Space Net collection, which includes high-resolution Digital Globe satellite photographs of five cities together with building footprints, is another source from which CNNs are used to segment images and extract building footprints. Despite its value, the spatial coverage and appropriateness of the Space Net dataset for classifier training are restricted. Remarkably, no other remote sensing dataset on this list possesses the enormous picture corpus required to develop a comprehensive image classification system.

Disadvantages

- Satellite imagery may be obstructed by cloud cover and tree canopies, limiting visibility.
- Ground truthing, employed for satellite image verification, is time-consuming and resource-intensive.

IV. PROPOSED SYSTEM

We provide a deep learning system designed specifically to categorize buildings and objects from high-resolution multi-spectral satellite images. Our method leverages satellite metadata to improve classification accuracy by combining a group ofa collection of post-processing neural networks and CNNs. Tested on the IARPA fMoW dataset, which comprises one million pictures in 60 categories, including a false detection class, our approach yields notable performance numbers. Our system has an accuracy of 0.83 and an F1 score of 0.797, and it performs very well in identifying 15 classes with 95% accuracy or better.Our technique has the potential to address a real-world problem raised at the beginning of this paper by efficiently searching through large volumes of satellite images to discover facilities or items of interest when paired with a detecting component. Our technology continually searches satellite photos to support law enforcement in identifying illicit operations such as unauthorized mining or fishing, as well as to aid natural disaster response teams map impacted areas. and make it easier for investors to keep an eye on developments in the agricultural or industrial sectors.

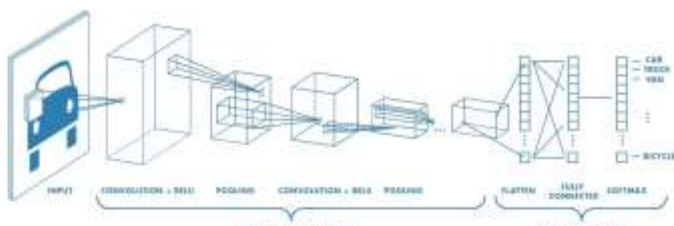

Fig. 1: The convolutional neural network's (CNN) architecture. During processing, the input image is sent through a series of image feature detectors.

Advantage

- Facilitates real-time monitoring and assessment of activities in any given location.
- Enables environmental impact analysis and historical data analysis for real-time reporting.

Modules

- Satellite image upload and detection
- Image feature extraction
- CNN Algorithm training
- Accuracy visualization
- Test image upload and classification

Functional Requirements

The system should handle the following:

- File management (filename)
- Deep learning accuracy tracking (deep_learning_acc)
- Classification (classifier)
- Coordinate tracking (X,Y)

Non-Functional Requirements

- **Usability:** User-friendly, ensuring ease of operation.
- **Security:** Implementation should adhere to security protocols, ensuring data integrity and user privacy.
- **Readability:** The system's interface and outputs should be easily understandable.
- **Performance:** The system should exhibit efficient execution and response times.
- **Availability:** Accessible and operational whenever required.
- **Scalability:** Scalable to accommodate potential future expansions or modifications.

Process Model

We adopt the SDLC umbrella model, encompassing stages such as Requirement Gathering, Analysis, Designing, Coding, Testing, and Maintenance. Each stage contributes refinement of the system, ensuring its robustness and effectiveness.

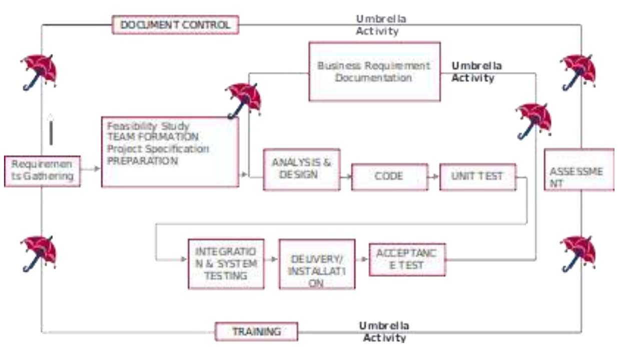

Fig. 2: SDLC Umbrella Model and Architecture.

Software Requirements Specification (SRS)

- **Requirement Study:** Conducted to identify and address project needs and challenges.
- **Feasibility Analysis:** Evaluates technical, operational, and economic feasibility of the project.
- **Operational Feasibility:** Ensures the system aligns with operational requirements and user needs.
- **Technical Feasibility:** Validates the system's technical viability and compatibility with existing infrastructure.

External Interface Requirements

- **User Interface:** Python Graphical User Interface for user interaction.
- **Hardware Interfaces:** Interaction facilitated through Python capabilities.
- **Software Interfaces:** Python environment for system operation.
- **Operating Environment:** Windows 10.

Hardware Prerequisites

- **Processor:** Intel Core i3 or i5 (minimum 1.1 GHz)
- **RAM:** a minimum of 256 MB
- **20 GB of Hard Disc Space -** Windows keyboard standard
- **Display:** SVGA

Software Prerequisites

- **Python 3.7.0 or Later;** - Windows 10 as the operating system; - OpenCV, Keras, Tensor Flow, Protobuf, H5py, Scikit-learn, Numpy, Pandas

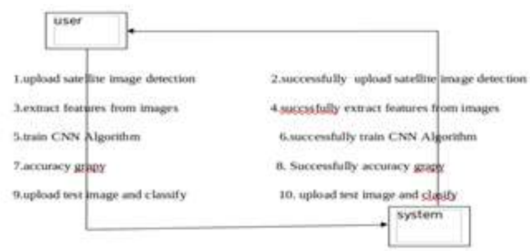

Fig. 3: An illustration that explicates the passage of data in the process.

Implementation

The main goals of the implementation phase are file conversion and user training. In-depth user training could be necessary, and depending on programming results, the system's initial parameters might need to be changed. The provision of basic operating procedures aids in the user's comprehension of system operations. Converting a new or updated system design into an operational one is the only complicated step in the system's overall implementation process.

Testing

Using test data that has been prepared, tests are conducted on individual modules to validate fields. This guarantees that everything works well together. To guarantee comprehensive testing, test data should cover a range of circumstances. Before real operation starts, testing aims to confirm that the system functions accurately and effectively.

System Testing and Module Testing

Testing is important in the field of information technology to ensure the reliability and readiness of a system before deployment. Various testing types are employed to guarantee software reliability. Logical and pattern testing are conducted to evaluate the program's execution and outcomes for different data sets.

Each module undergoes individual testing to detect and correct errors without affecting other modules. The system's modules sequentially, starting from the smallest and lowest-level modules and progressing upwards. For instance, modules such as job classification and resource allocation are tested separately to ensure efficient system performance.

Integration Testing:
After module testing, integration testing is performed to identify and rectify errors that may occur when linking modules. All modules are interconnected and tested to ensure correct functioning of the entire system. Integration testing confirms the accurate mapping of jobs with resources.

Acceptance Testing
The system goes through one last round of acceptability testing after users verify its accuracy and functioning. This test verifies that the system satisfies the initial requirements, aims, and objectives set out during analysis. Acceptance testing saves time and money by confirming that the system is ready for use without requiring users to complete real tasks. These test cases verify the functionality of the system, making it possible to find and fix any problems prior to deployment.

V. RESULTS

Fig. 4: Entity diagram for Uploading the data set is shown below

Fig. 5: Entity diagram showing CNN training is completed and we got its accuracy as 91%.

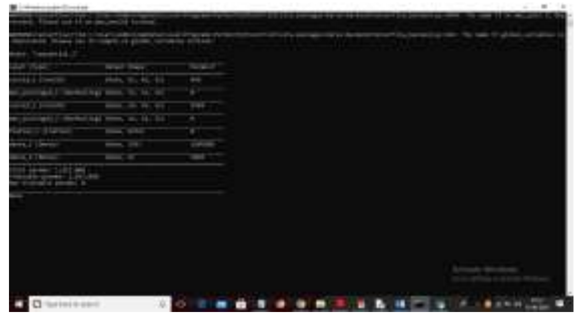

Fig. 6: Simulation results are obtained for all the layer details of CNN .

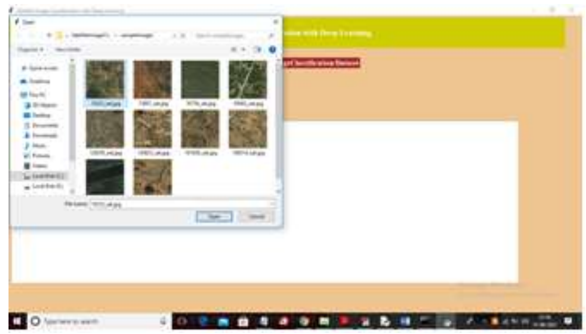

Fig. 7: Entity diagram for selecting and uploading '10233_sat.jpg'

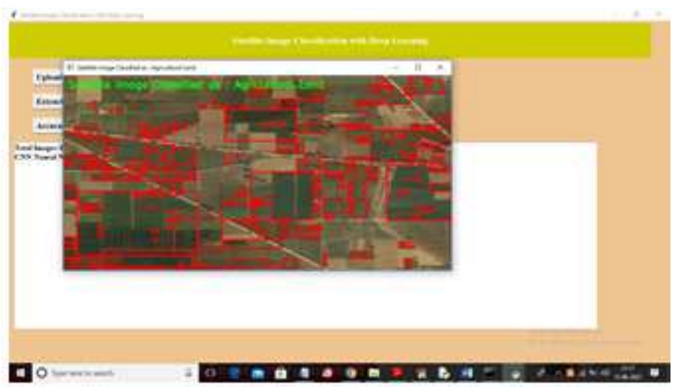

Fig. 8: Entity diagram showing green color text where we can see image classified as 'Agriculture Land' and all agriculture area is surrounded with red color bounding boxes

VI. CONCLUSION

In conclusion, by integrating a CNN ensemble with post-processing neural networks that leverage satellite information, our deep learning system accurately classifies objects and buildings from high-resolution multi-spectral satellite photos. More precisely, our system performs well on the IARPA fMoW dataset, which has one million images split into 63 classes, including a false detection class, with an accuracy of 0.83 and an F1 score of 0.797. Its 95% accuracy rate in identifying 15 classes is outstanding; in the fMoW Top Coder competition, it is 4.3% faster than the Johns Hopkins APL model.

Moreover, by including a detecting component, our system demonstrates its ability to effectively search through enormous volumes of satellite photos for buildings or objects of interest. This functionality fixes the problems that were previously discussed in this article. Furthermore, our system can help law enforcement detect illegal mining activities or fishing vessels, help natural disaster response teams map hurricane damage or mudslides, and help investors track crop growth or oil well development more precisely and effectively through continuous satellite imagery monitoring.

REFERENCES

1. Volpi, M. "Deep Learning for Satellite Imagery Analysis: A Survey," arXiv:1802.01344, February 2018.
2. In 2017, N. Lebatteux, A. Bazi, and J. Zerubia published "Land-use classification in remote sensing images using convolutional neural networks," 9, no. 3, p. 223.
3. In "Classification of high-resolution remote sensing imagery using deep learning," S. Basu, A. Bhowmik, and S. Bandyopadhyay, vol. 56, no. 1, pp. 588-598, Jan 2018.
4. In 2017, Chin, Fu, and Yong published "A deep learning approach for urban land use classification using street view images," which was published in 9, no. 5, page 505.
5. M. Ahmad, "Deep learning based semantic classification of remote sensing data: A systematic review," in Remote Sensing Applications: Society and Environment, vol. 14, Dec. 2019, pp. 147–167.
6. "Deep learning algorithms for remote sensing applications: A review," S. M. Ali and H. A. Jalab, 12, no. 1, p. 109, Jan 2020
7. In August 2020, J. Park and S. Kim published "Fusion of Hyperspectral and LiDAR Data for Urban Land Cover Classification using DL," in the IEEE Journal of Selected Topics in Applied Earth Observations and 13, pp. 3873–3885.
8. Transfer learning in remote sensing for agricultural land cover classification: a comparative study by A. Gupta, R. Sharma, and V. Singh, Remote Sensing Letters, vol. 11, no. l. 14, no. 1, p. 016510, Jan 2020.
9. "Hyperspectral Image Classification Using Deep Learning: A Comparative Study of CNN and LSTM Networks," S. Roy, A. Banerjee, and P. Gupta, vol. 14, no. 1, p. 016510, Jan 2020.
10. IEEE 2020 International Seminar on Applications of Intelligent Technology
11. Creating Creative Business Plans for Programmes in Small Spaces Dit S by Spin-Outs by Monge (2018)