DeepSat: Revolutionizing Computer Vision in Satellite Imagery with “Advanced Learning Frameworks”

Prakash Kumar
Department of Computer Science
Lovely Professional University,
Phagawara,Punjab,144411
prakash.12019155@lpu.in
Reg no : 12019155

Dr. Usha Mittal
Department of Computer Science
Lovely Professional University
Phagawara,Punjab,144411
Ushamittal123@gmail.com

***Abstract*— "Satellite imagery plays a pivotal role in various critical applications, such as disaster response, law enforcement, and environmental monitoring. These applications necessitate the manual identification of objects and facilities within the imagery. Given the vast geographic areas to be surveyed and the limited availability of human analysts, automation becomes imperative. However, conventional object detection and classification algorithms have proven to be inadequate and unreliable in addressing this challenge. Computer vision algorithms, a subset of machine learning techniques, hold significant promise for automating such tasks. In particular, computer vision algorithms leveraging techniques such as feature extraction and image segmentation offer potential solutions. In this paper, we propose a computer vision system designed to classify objects and facilities from high-resolution, multi-spectral satellite imagery. Our research employs computer vision algorithms applied to the IARPA Functional Map of the World (fMoW) dataset, categorizing objects into 63 different classes. The system is constructed as a combination of various computer vision algorithms, including feature extraction, object recognition, and image processing techniques. Furthermore, we integrate satellite metadata with image features to enhance classification accuracy. The implementation is carried out using Python, utilizing popular computer vision libraries such as OpenCV. The system operates on a Linux server equipped with NVIDIA graphics hardware for optimized performance. The overall accuracy of the system is 83%, while the F1 score stands at 0.797. Impressively, it accurately classifies 15 of the classes with precision rates of 95% or higher. This research signifies the potential of computer vision techniques in the realm of satellite imagery analysis, presenting a viable solution to automate object and facility recognition in this context."**

Keywords—artificial intelligence; AI; Classification accuracy; machine learning; image understanding; recognition; classification; satellite imagery

I. INTRODUCTION

Computer vision is a specialized field within artificial intelligence that's dedicated to creating algorithms capable of interpreting and processing visual data, such as images and videos. In this context, we explore the synergy of advanced computer vision techniques and their significant impact on image analysis. These models operate by processing data through multiple layers of abstraction, enabling them to recognize intricate patterns and features within images [1].

One prominent facet of image analysis involves the use of convolutional neural networks (CNNs), which are a subset of deep learning techniques but are fundamentally rooted in computer vision. These networks consist of multiple layers designed to identify intricate visual patterns and features within images. In essence, CNNs serve as a powerful tool for object detection and classification within the domain of computer vision. And harnessing the computational power of graphical processing units (GPUs). Since 2012, CNN-based algorithms have consistently outperformed other methods in challenges like the ImageNet Large Scale Visual Recognition Challenge, which involves detecting and classifying objects in photographs [2]. This success has prompted major technology companies, including Google, Microsoft, and Facebook, to deploy CNN-based products and services in various applications [1].

Over the years, computer vision has witnessed a remarkable progression in the complexity and effectiveness of image analysis algorithms. We have seen the emergence of extensive networks and intricate models tailored to a range of applications. For instance, VGG introduced a model comprising 16 layers, offering a deeper understanding of visual content [7]. Google's Inception, with its 22 layers, further demonstrated the possibilities of feature extraction in computer vision [8]. Subsequent iterations of the Inception model have added even more layers, providing more refined feature extraction capabilities [9].

What sets computer vision techniques apart is their capacity to automatically detect and interpret image features without the need for manual feature engineering. Unlike older methodologies like Scale-Invariant Feature Transform (SIFT) and Histogram of Oriented Gradients (HOG), computer vision algorithms, particularly CNNs, are adept at autonomously identifying relevant features and patterns during their training process.

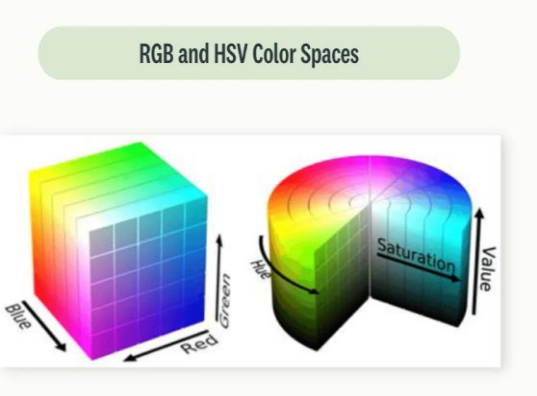
These advancements in computer vision, along with the use of sophisticated image analysis techniques, have led to a profound shift in our understanding of visual data and its practical applications across various domains. By leveraging these computer vision methodologies, we are equipped to

unlock a deeper and more detailed understanding of visual content, enabling more accurate and effective image analysis.

In order to make it possible for computers to process image data, they have to be converted into a numerical format.

Every time an image is captured from a hardware device (e.g. photo camera) it is therefore stored as a structure of numbers following a specific color space. Two of the most famous color spaces are RGB (Red, Green, Blue) and HSV (Hue, Saturation, Value).

Using RGB for example, an image would be stored as a 3-dimensional array with each channel representing one of the 3 primary colors (Figure 1). Combining the 3 channels together, we would then be able to recreate on screen the original image.



After an image is stored in a system, it can undergo meticulous processing to enhance specific features that are pivotal for enabling discrimination between various potential outcomes within Machine Learning systems.

In the realm of traditional Computer Vision methodologies, this process typically encompasses the utilization of image preprocessing techniques, including point/group operators (e.g., Intensity Normalization, Histogram Equalization, Gaussian Averaging), as well as the extraction of distinctive features through methods like Block-Based Features and Local Features (e.g., Dense SIFT, SIFT)

This would then ultimately enable us to use Bag of Visual Words and a standard ML algorithm in order to make predictions.

The advent of cutting-edge model architectures like Transformers and Convolutional Neural Networks (CNNs) has ushered in a new era where the emphasis on meticulous image preprocessing steps has been substantially reduced. These advanced models have the capability not only to predict outcomes effectively but also to autonomously determine the optimal features to extract, thus achieving outstanding results.

II. Problem

Searching for objects, facilities, and events of interest in satellite imagery is an important problem. Law enforcement agencies seek to detect unlicensed mining operations or illegal

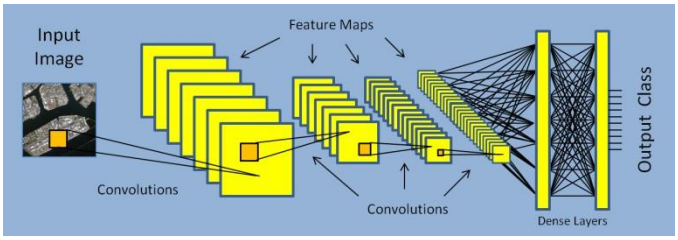


Fig. 2. The structure of a convolutional neural network (CNN). The input image is passed through a series of image feature detectors.

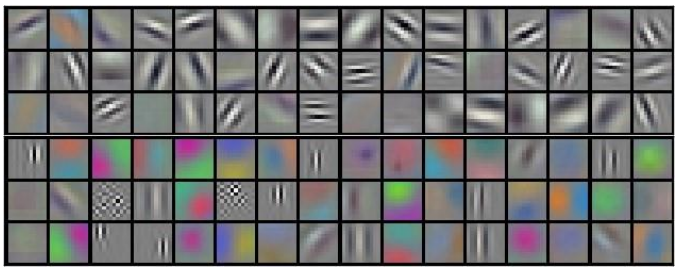


Fig. 3. Examples of the image feature detectors that a CNN might “learn” during its training [6].

fishing vessels, disaster response teams wish to map mudslides or flooding, and financial investors seek better ways to monitor agricultural development or oil well construction. Because the geographic expanses to be covered are great and the analysts available to conduct the searches are few, automation is required. Yet traditional object detection and classification algorithms are too inaccurate and unreliable to solve the problem. What is needed is a deep learning system that can recognize and label objects and facilities automatically, as illustrated in Fig. 3.

"In the domain of computer vision applied to satellite imagery, the integration of deep learning methods has encountered hurdles. A prominent issue pertains to the intricate preprocessing of satellite image data to align with the demands of computer vision algorithms. Despite several recent endeavour’s, achieving optimal preprocessing for computer vision tasks remains a challenging endeavor in this context."

Introduction to Satellite Imagery

Satellite imagery constitutes a crucial category of geospatial data, procured through the deployment of satellites and aircraft to gather valuable insights about the Earth. This data acquisition involves the utilization of a diverse array of active and passive sensors.

Geospatial data is commonly categorized into two primary formats: Rasters and Vectors.

Raster data is depicted as a grid of pixels, thus maintaining a fixed spatial resolution. For instance, within the RGB color space, a single image can be encoded using three channels or bands. Raster data is conventionally saved using formats like TIFF, PNG, JPG, and similar file types.

In contrast, Vector data serves the purpose of recreating real-world geometries using elements such as points, lines, polygons, and more. It also encompasses additional metadata concerning the represented objects. Leveraging their inherent mathematical properties, Vector data permits zooming in and out without any loss of resolution. Such data is commonly stored in file formats such as SVG and Shapefiles.

To keep processing times reasonable, CNNs require relatively small fixed-size images. For example, ResNet [10] and DenseNet [11] work with 224x224-pixel images, while Inception [8,9] accepts images of size 299x299. The standard practice in deep learning is to crop and warp the images to the required size [14]. For ordinary photographs, these operations preserve important image features. Such is not the case for satellite images, however, because objects and facilities can be much larger than objects in ordinary photographs. Airports and shipyards, for example, can be tens of thousands of pixels in size. When such large images are resized to a much smaller size of 224x224 or 299x299, small details are lost. These details might include important distinguishing features such as airplanes on a tarmac or container cranes in a shipyard. In addition, there is no way to crop such an image to a smaller size in an effort to preserve small features without losing much of the image. Other complications that plague satellite images as opposed to ordinary photographs include the extremely large aspect ratios of objects such as runways and chicken barns, the wider range of object orientations such as upside-down buildings, and obscuration by clouds as shown in Fig. 4. These conditions make it a challenge to prepare satellite imagery properly for input to CNNs.

Fig. 4. The results of object and facility recognition with deep learning applied to satellite imagery. The unlabeled boxes are false detections.

Fig. 5. Objects in satellite images are subject to more extreme viewing conditions than objects in ordinary photographs. They can be very large or very small, long and narrow, upside down, or obscured by clouds.

Satellite Imagery is ubiquitous and used in many different applications such as: meteorology, military planning, environmental assessment, mapping, etc. Successfully combining Satellite Images with AI systems can therefore potentially result in large-scale optimization across different research areas

Computer Vision and Satellite Imagery

AI applications in satellite imagery can be categorized into one-level and multi-level applications.

In one-level applications, the primary focus revolves around utilizing satellite imaging to execute a particular task. This may involve precise detection of various objects within an image. Notably, this task can be considerably more intricate compared to similar tasks involving everyday images, mainly due to the smaller relative size of objects in aerial imagery when compared to the overall image dimensions.

Multi-level applications involve a more intricate process, where information is initially extracted from aerial imagery using the one-level approach. This information is then amalgamated with additional features to empower more sophisticated Machine Learning applications, such as smart city planning.

Various computer vision techniques find application in the analysis of aerial imagery, including:

Object Detection: Identifying different objects within images, often achieved through techniques like Object Detection using YOLO (You Only Look Once) and the creation of bounding boxes.

Image Segmentation: Dividing images into their core constituents by distinguishing backgrounds from foregrounds with instance segmentation techniques.

Image Classification: Assigning images to distinct categories based on their content, which is a fundamental task in image analysis.

Feature Matching: Employed to determine whether two different images have captured the same object, this technique plays a critical role in recognizing and tracking objects across various images

Challenges of Training Computer Vision Models on Satellite Imagery

The field of satellite imagery analysis presents several prominent challenges that researchers and practitioners must contend with:

Varied Image Quality: Satellite images often exhibit inconsistent quality owing to factors such as atmospheric disturbances, sensor noise, and fluctuations in the satellite's distance from Earth's surface. Challenges such as cloud cover, smog, haze, and varying resolutions across images contribute to image quality variations.

Geometric and Radiometric Distortions: Aerial perspective introduces geometric distortions due to differing altitudes and angles from which images are captured. Additionally, variations in sun angle, sensor view angle, and atmospheric conditions can lead to radiometric distortions.

Scale Variation: Objects of interest can manifest at vastly different scales in satellite images, influenced by the satellite's altitude and camera angle. This scale variability presents difficulties in detecting and analyzing objects, especially small ones.

Temporal Changes: Satellite images of the same location captured at different times can exhibit variations due to changes in lighting, weather conditions, seasonal vegetation, or human activity. Maintaining temporal consistency across analysis remains a substantial challenge.

Lack of Labeled Data: Supervised learning approaches necessitate the labeling of images or segments with their correct interpretations. Labeling aerial imagery is a labor-intensive task that requires expert knowledge.

Large Volume of Data: Satellite image datasets can be exceptionally large, demanding extensive computational resources for storage and processing.

Multi-spectral Bands: Satellite images often encompass multiple spectral bands beyond the conventional red, green, and blue. While these additional bands can enhance model performance, they introduce complexity in model development.

Data Availability and Privacy Concerns: In specific regions, access to high-resolution images may be restricted or unavailable due to national security or privacy concerns.

Domain Adaptation: Models trained on one set of images, such as urban landscapes, may exhibit poor performance when applied to different sets of images, such as rural landscapes or distinct geographical locations, due to distribution shifts.

Addressing these challenges in the context of satellite imagery analysis requires innovative solutions and a deep understanding of the unique characteristics of such data.

The role of data labeling tools in training computer vision models

Before we delve into our case studies, it's crucial to underscore the significance of data labeling tools they provide a robust suite of image annotation capabilities that play a vital role in the training of computer vision models. More than just basic annotation, they empowers engineers to develop plugins that integrate with their models, facilitating the scalability of high-quality image annotation tasks. These tasks encompass programmatic quality assurance, consensus resolution, and

parallel labeling, making it a comprehensive solution that elevates the entire lifecycle of model development.

Use case: Object Classification using the Pavia University scene dataset

To illustrate a one-level Computer Vision application, we will guide you through a practical example of classifying various objects in a satellite image using the Pavia University scene dataset. This dataset contains image data obtained through the ROSIS sensor during a flight inspection conducted over Pavia, Italy. The imagery boasts a geometric resolution of 1.3 meters, making it a valuable resource for our classification task

https://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes (this is the link for the dataset used to test and train the model)

Data Preprocessing steps

First and foremost, we'll import all the essential libraries for this task. In the Colab environment, you can use libraries that are readily available. Let's go with PyTorch, which is a popular choice for deep learning tasks in Colab.

```
import torchvision
import torch.nn as nn
import torch.optim as optim
import torchvision.transforms as transforms
from torch.utils.data import DataLoaderscribed:
```

To import the aerial images dataset and convert it into a tabular format in Colab, you can use various Python libraries. One commonly used library is NumPy. Here's how you can perform the data conversion and preprocessing as described:

Finally, the different image bands are stacked together in order to make it easier to create some visualizations. In Figure 6, we can then see our starting data and our ground truth with the image data classified into 9 different categories.

Fig 6 : The results of object and facility recognition

We are now ready to go deeper into our own data and plot 3 sample bands in order to get an idea of what they represent and how they can be used together in order to make predictions (Figure 7).

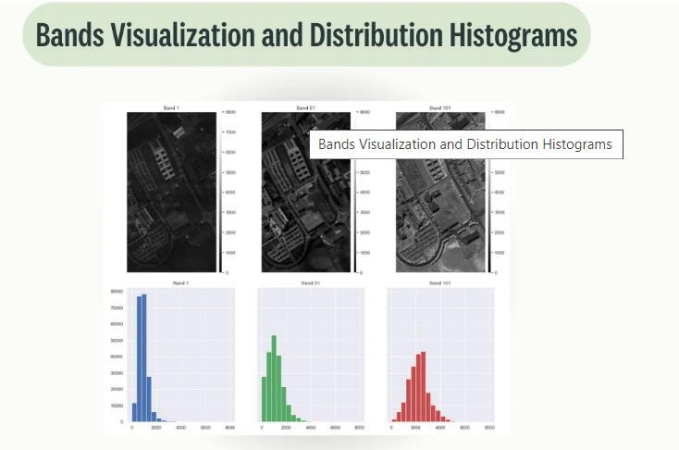


Fig. 7. Objects in satellite images are subject to more extreme viewing conditions than objects in ordinary photographs. Showing bands visualization with Distribution Histogram

Machine learning

At this point, we can then try to design a standard Decision Tree classifier in order to predict the different object classes in the image. Using just this simple model, an accuracy score of 88.3% is registered on our test set.

Once trained the model, we can then use it to make predictions for the different pixels on the entire image data and plot the results to get some form of visual feedback on how useful our model can be in classifying buildings, etc. (Figure 8).

Fig. 8. Showing Decision tree classifier prediction using meachine learning model

Convolutional Neural Network

Now that we have created a machine learning good baseline to start from, we can then try to use more advanced Deep Learning based approaches such as Artificial Neural Networks and Convolutional Neural Networks.

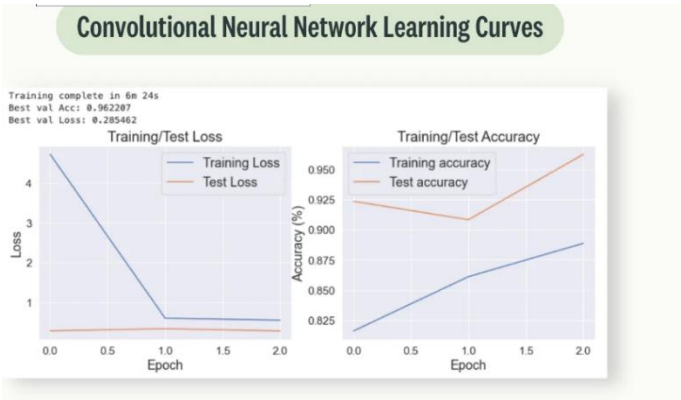
In order to take advantage of the spatial resolution of our data we then need to construct image patches and padding our edges.

In order to make the image generation process faster, we can then apply Principal Component Analysis (PCA) to reduce the dimensionality of our input data. If you are interested in learning more about PCA and other Feature Extraction techniques

We are now ready to define our CNN. In this case, we start with applying 3-dimensional and 2-dimensional convolutions to extract features from the input data and finally flatten it to make predictions.

In order to better understand how our model is learning during training time, can then make a helper function to visualize the learning curves for both training and validation steps.

At this point, we are now ready to wrap everything into our training loop function and get back the best model weights recorded as part of the process.Finally, we are now ready to define our loss function, preferred optimizer and trigger our model training. Using this approach, an accuracy rate of 96.2% is registered on our validation data, outperforming therefore our baseline model (Figure 9).

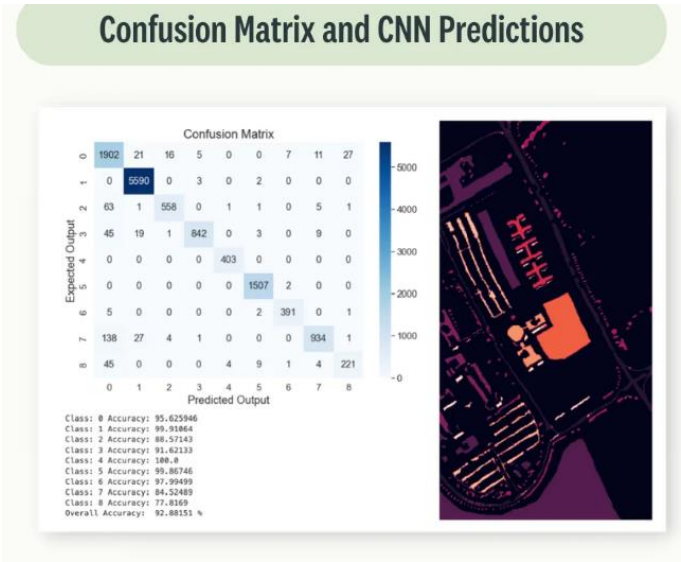


Validation

Although the accuracy rate is looking really promising, is always important as part of an analysis to visualize also the accuracy rate for each individual class and the overall confusion matrix. In fact, if our class distribution is highly imbalanced we might then still be able to make a highly performant model by always betting on the majority class.

Ultimately, we can now make use of our trained model to predict our full ground truth image.

Inspecting Figure 10, we can then successfully validate that our model is able to perform relatively well for each different class and that the predicted image looks much closer to the original ground truth.



Use case: Predicting Crop Yields using Satellite Imagery

An example of multi-level applications can instead be Crop Yields prediction. In this sort of use case, we aim to predict crop yields (e.g., corn, wheat) in agriculture fields before harvest time to enable better planning and decision-making for farmers, commodity traders, and policy-makers (especially considering potential climate change effects).

In order to make these sorts of predictions accurately image data might in fact not be enough and other sources of information such as measurements of rainfalls, temperatures, gps coordinates, etc. are vital in order to make accurate predictions.

Data Collection

For the image collection process, we need to acquire aerial imagery covering our region of interest throughout the growing season. These images should ideally have different spectral bands, including visible light and near-infrared, which are useful for assessing plant health. This sort of data can be sourced from providers like Landsat, Sentinel, or commercial providers.

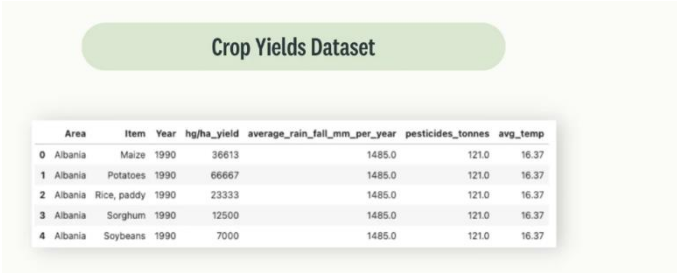
Data Preprocessing

For brevity, in this case, we assume we have already been given image data, created a classification model to classify the different types of crop, and integrated this information with other data sources in a tabular format to have everything pre processed.

In this case, our labels will be the actual crop yields for each region, which in real life might be obtained from agricultural agencies or surveys.

In order to perform this process at its best radiometric and geometric corrections should be applied to the images. Pixel values should have been normalized and spectral bands combined/selected to create vegetation indices, such as the Normalized Difference Vegetation Index (NDVI), which is widely used to assess plant health and growth.

For this example, we are going to use Kaggle Crop Yield Prediction Dataset (Figure 11 and 12)



	Area	Item	Year	hg/ha_yield	average_rain_fall_mm_per_year	pesticides_tonnes	avg_temp
0	Albania	Maize	1990	36613	1485.0	121.0	16.37
1	Albania	Potatoes	1990	66667	1485.0	121.0	16.37
2	Albania	Rice, paddy	1990	23333	1485.0	121.0	16.37
3	Albania	Sorghum	1990	12500	1485.0	121.0	16.37
4	Albania	Soybeans	1990	7000	1485.0	121.0	16.37

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.decomposition import PCA
from sklearn.tree import DecisionTreeRegressor
from sklearn.metrics import r2_score

df = pd.read_csv('yield_df.csv')
df = df.iloc[: , 1:]
df.head()
```

In this example, our dataset contains crop data from a large number of countries with India covering on its own 28.3% of the available data.

Discussion

Remember that this use case is complex, and other factors might influence crop yields, like soil type, farming practices, etc. Integrating these additional data sources could improve our model's accuracy. Also, crop types and growth stages vary spatially and temporally, so the model should ideally be retrained or fine-tuned for different regions and seasons.

As part of this article, we explore how Computer Vision can be used in order to get more value out of aerial imagery and provided different use case applications (e.g. natural disaster prediction). Although, as this area of research keeps growing, it is always important to keep updated with new techniques and libraries to streamline your workflows.

II. Conclusion

We have presented a deep learning system that classifies objects 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 IARPA fMoW dataset of one million images in 63 classes, including the false detection class, the system achieves an accuracy of 0.83 and an F_1 score of 0.797. It classifies 15 classes with an accuracy of 95% or better and beats the Johns Hopkins APL model by 4.3% in the fMoW TopCoder challenge.

Combined with a detection component, our system could search large amounts of satellite imagery for objects or facilities of interest. 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.

ACKNOWLEDGEMENTS

We would like to express our heartfelt gratitude to Dr..Usha Mittal for her invaluable contributions and unwavering support throughout this project. Her expertise, guidance, and dedication have been instrumental in its success. We are deeply thankful for her mentorship and collaboration.

REFERENCES

[1] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” Nature, vol. 521, pp. 436-444, 28 May 2015.

[2] “Large Scale Visual Recognition Challenge (ILSVRC) – ImageNet,” ImageNet, http:// www.image-net.org/challenges/LSVRC.

[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] Y. LeCun et al., “Backpropagation applied to handwritten zip code recognition,” Neural Computation, vol. 1, no. 4, pp. 541-551, 1989.

[6] A. Krizhevsky, I. Sutskever, and G. Hinton, “ImageNet classification with deep convolutional neural networks,” NIPS: Neural Info. Proc. Sys., Lake Tahoe, Nevada, 2012.

[7] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” arXiv 1409.1556, Sep 2014.

[8] C. Szegedy et al., “Going deeper with convolutions,” arXiv 1409.4842, Sep 2014.

[9] C. Szegedy et al., “Rethinking the Inception Architecture for Computer Vision,” IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[10] K. He et al., “Deep residual learning for image recognition,” arXiv 1512.03385, Dec 2015.

[11] G. Huang, “Dense connected convolutional neural networks,” IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[12] “TensorFlow: An open-source software library for machin intelligence,” TensorFlow, https://www.tensorflow.org/.

[13] F. Chollet at al., “Keras”, GitHub, 2017, https://github.com/fchollet/keras.

[14] K. He, X. Zhang, S. Ren, and J. Sun, “Spatial pyramid pooling in deep convolutional networks for visual recognition,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 37, no. 9, pp. 19041916, Sep 2015.

[15] “UC Merced Land Use Dataset,” University of California, Merced, http://weegee.vision.ucmerced.edu/datasets/landuse.html.

[16] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," Proc. 18th ACM SIGSPATIAL International Symposium on Advances in Geo. Info. Sys., pp. 270-279, 3-5 Nov 2010.

[17] Y. Liang, S. Monteiro, and E. Saber, “Transfer learning for highresolution aerial image classification,” IEEE Workshop Applied Imagery Pattern Recognition (AIPR), Oct 2016.

[18] M. Castelluccio, G. Poggi, and L. Verdoliva, “Land Use Classification in Remote Sensing Images by Convolutional Neural Networks,” arXiv 1508.00092, Aug 2015.

[19] 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, pp. 549-553, Apr 2017.

[20] “SpaceNet on AWS,” Amazon.com, https://aws.amazon.com/publicdatasets/spacenet/.

[21] E. Chartock, W. LaRow, and V. Singh, “Extraction of Building Footprints from Satellite Imagery,” Stanford University Report, 2017.

[22] G. Cheng, J. Han, and X. Lu, “Remote Sensing Image Scene Classification: Benchmark and State of the Art,” Proc. IEEE, vol. 105, no. 10, pp. 1865-1883, October 2017.

[23] “Functional Map of the World Challenge,” IARPA, https://www.iarpa.gov/challenges/fmow.html.

[24] “Marathon Match: Functional Map of the World,” TopCoder, https://community.topcoder.com/longcontest/?module=ViewProblemStatement&rd=16996&compid=57158.

[25] “Earth on AWS: Functional Map of the World,” Amazon.com, https://aws.amazon.com/earth/.

[26] G. Christie. N. Fendley, J. Wilson, and R. Mukherjee, “Functional map of the world,” arXiv 1711.07846, 21 Nov 2017.

[27] F. Chollet, “Xception: deep learning with depthwise separable convolutions,” arXiv 1610.02357, Oct 2016.

[28] C. Ju, A. Bibaut, and M. van der Laan, “The relative performance of ensemble methods with deep convolutional neural networks for image classification,” arXiv 1704.01664, April 2017.

[29] “Applications”, Keras, https://keras.io/applications/.

[30] F. Yu, “CNN Finetune”, Github, 2017, https://github.com/flyyufelix/cnn_finetune.

[31] A. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, “CNN features off-the-shelf: an astounding baseline for recognition,” IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), pp. 512-519, 2014.

[32] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, “How transferable are features in deep neural networks?” Proc. 31st Int. Conf. Machine Learning, vol. 32, pp. 647-655, 2014.