24678 - Computer Vision Final Project Report

Title: Object Extraction from High Resolution Satellite Images

Team Chicago

Contents

Abstract:

The motivation behind undertaking this project on object extraction in satellite images stems from the increasing availability of high-resolution satellite imagery and the growing need for automated analysis of these vast datasets. Satellite imagery provides a wealth of valuable information for various applications, including urban planning, environmental monitoring, disaster management, and agricultural assessment. However, manually extracting objects of interest from these images is a time-consuming and labour- intensive task. This project aims to classify the contents of a satellite image with the help of topics covered during the lectures and deep learning models. We aim to develop a robust solution for automating classification of objects in satellite images. Below shows a graph that tells how the number of satellites are increasing every year. With the help of this graph we can tell that automating the process of identifying features from a large lot of pictures taken by the satellites is really helpful.

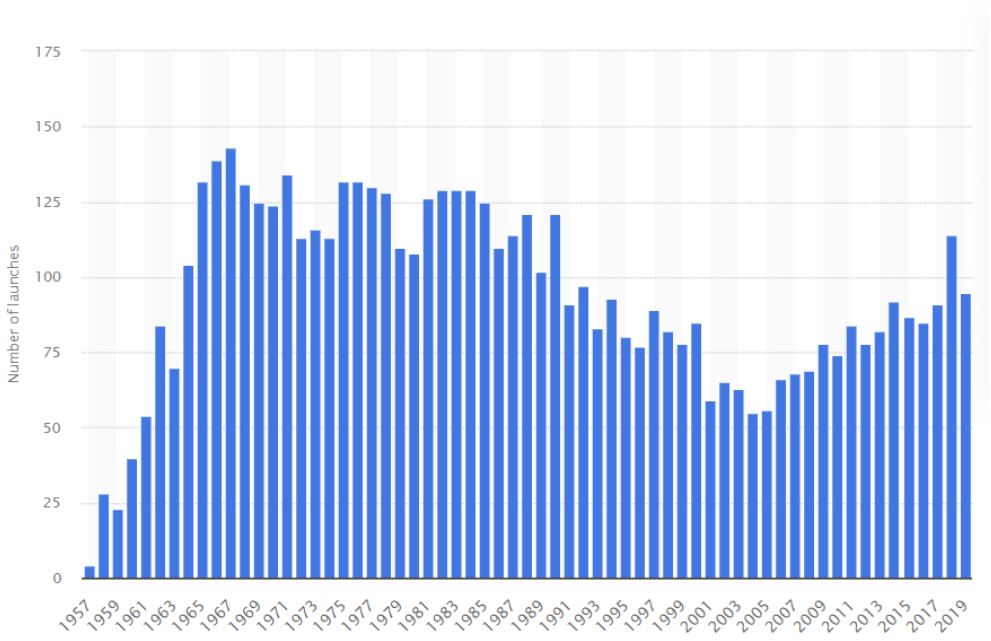


Fig. Number of satellite launches vs year launched.

Methodology:

Dataset Overview:

- Utilized the DOTA dataset, which consists of 1500 train images, 400 validation images, and 400 testing images.
- Each image has multiple bounding boxes (Oriented Bounding Boxes – OBB) with associated category labels.

Fig. Dataset overview

Visualization and Data Inspection:

- Developed a code to visualize and read bounding boxes on top of the images for better understanding.
- Extracted bounding box coordinates for training and validation datasets.
- Formatted the data into a numpy array for each image, where each row contains [x1, y1, x2, y2, x3, y3, x4, y4, category] for each bounding box.
- Handled cases where there are multiple bounding boxes per image.

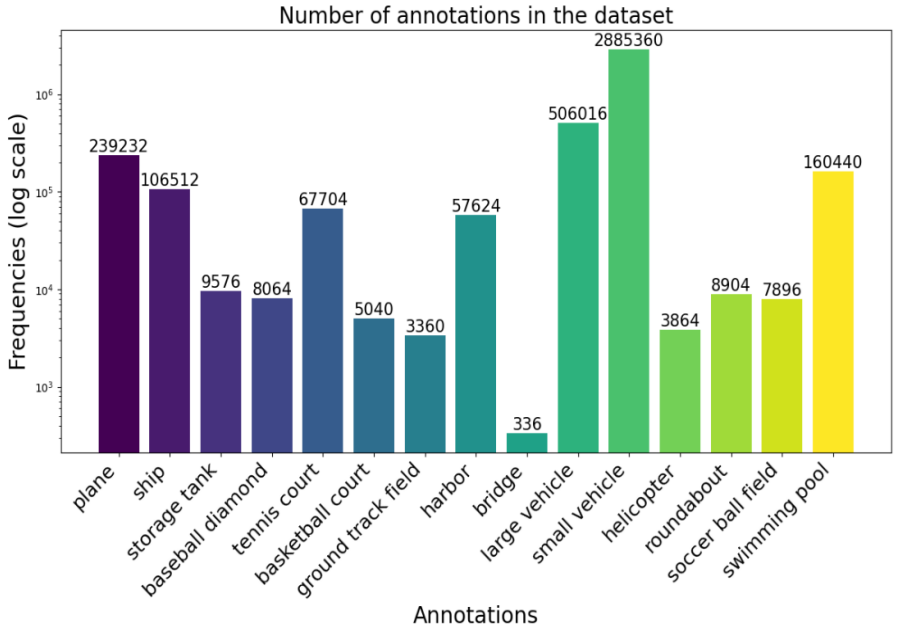


Fig. Frequency of classes vs classes

Data Pre-processing:

1. Image Splitting:

- Due to large image sizes (ranging from 5000 x 5000 to 20000 x 20000 pixels), implemented an image splitting step to reduce computational complexity.
- Divided the images into smaller, manageable tiles of 9, ensuring that each tile contains relevant objects and maintains context.

2. Resizing:

- Resized all split images to a standardized size (e.g., 1024 x 1024 pixels). This ensures a consistent input size for the model and simplifies computation.

3. Coordinate Normalization:

- Normalize the coordinates of the bounding boxes:
- Divided x-coordinates by the width of the image.
- Divided y-coordinates by the height of the image.
- Normalized category labels

Model Selection:

- Choose a YOLO (You Only Look Once) model for object detection. YOLO models are known for their real-time processing and efficiency.
- Initially, we have build a simple neural network that takes in input as a csv file. The cells in the CSV file consist of path of the image, the bounding box coordinates and the target class. The problem that this model had was firstly, the output of the model was only one bounding box whereas the images had more than one bounding box. Secondly, the size of the csv file created was around 190 thousand rows long, making this training of model on this dataset extremely difficult.
- All these problems made the YOLO model a viable solution.

Pipeline:

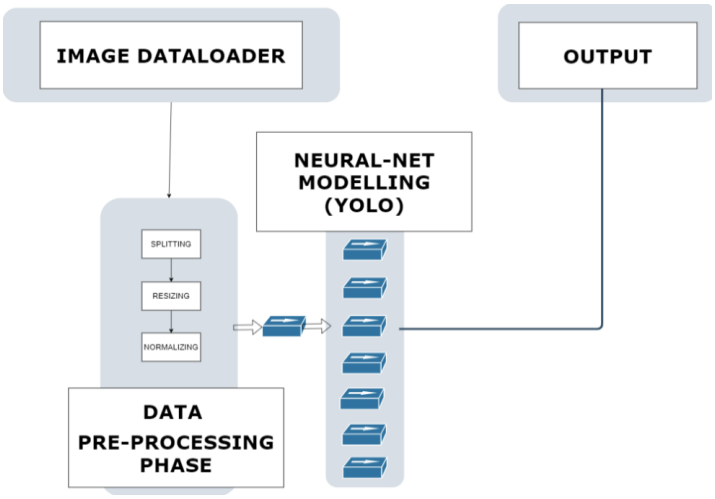


Fig. Model architecture followed for this project

Performance Metrics:

- The performance metric used is for this project is **Mean Average Precision (MAP).**
- To get a clear understanding of MAP we need to understand Intersection of Union (IOU).
- IoU measures the overlap between the predicted bounding box and the ground truth bounding box of an object, and is calculated as the ratio of the intersection area to the union area of the two boxes.

Model hyper parameters:

- Train Box Loss: 1.628

  This is the training loss associated with the bounding box coordinates. It measures how well the model is predicting the locations of the bounding boxes around objects in the training images. A lower box loss indicates better accuracy in predicting the object's position.

- Train Classification Loss: 1.147

  This is the training loss associated with the classification of objects. It measures how well the model is assigning the correct class labels to the objects in the training images. A lower classification loss indicates better accuracy in predicting the object classes.

- MAP (Mean Average Precision): 62%

  Mean Average Precision is a commonly used evaluation metric for object detection models. It combines precision and recall across different confidence thresholds. A higher MAP indicates better overall performance of the model on the validation or test dataset.

- Epochs: 100

  This indicates the number of times the entire training dataset has been processed by the model during training. Training for 100 epochs suggests that the model has iteratively seen the entire dataset 100 times in the training process.

- Batch Size: 16

  During training, the dataset is divided into batches, and each batch contains 16 images. The model's parameters are updated based on the average loss calculated over these 16 images. Smaller batch sizes may lead to more frequent updates but may be computationally expensive, while larger batch sizes can be more computationally efficient but may slow down convergence.

Training:

- Split the pre-processed dataset into training and validation sets.
- Trained the YOLO model using the resized and normalized images with associated bounding box coordinates and category labels.

Testing and Evaluation:

- Evaluated the trained model on the testing dataset.
- Measured performance metrics to assess the model's accuracy.

Downstream Post processing:

- After fine-tuning the model can perform specific downstream tasks.
- One such application is identifying the traffic in a particular region on the basis of the number of small vehicles and large vehicles in a region.
- This can be accomplished by using the trained weights of the model and making slight changes based on the application area

Results Visualization:

- ***Pre-Processing:***

Fig. Input image

- ***Loss Curve:***

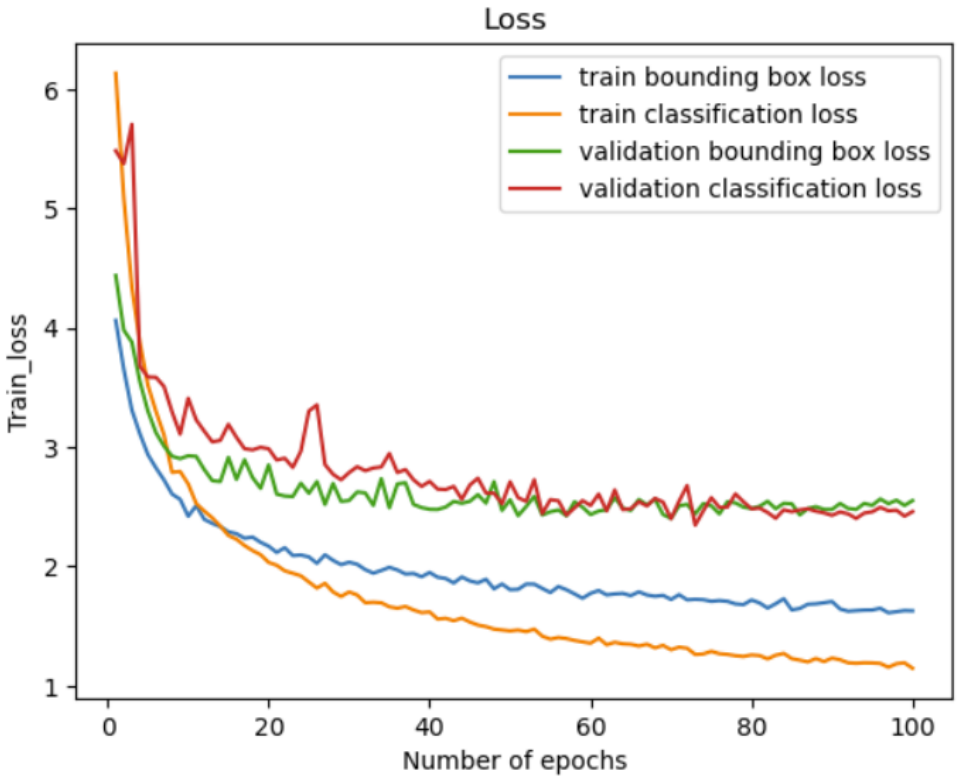
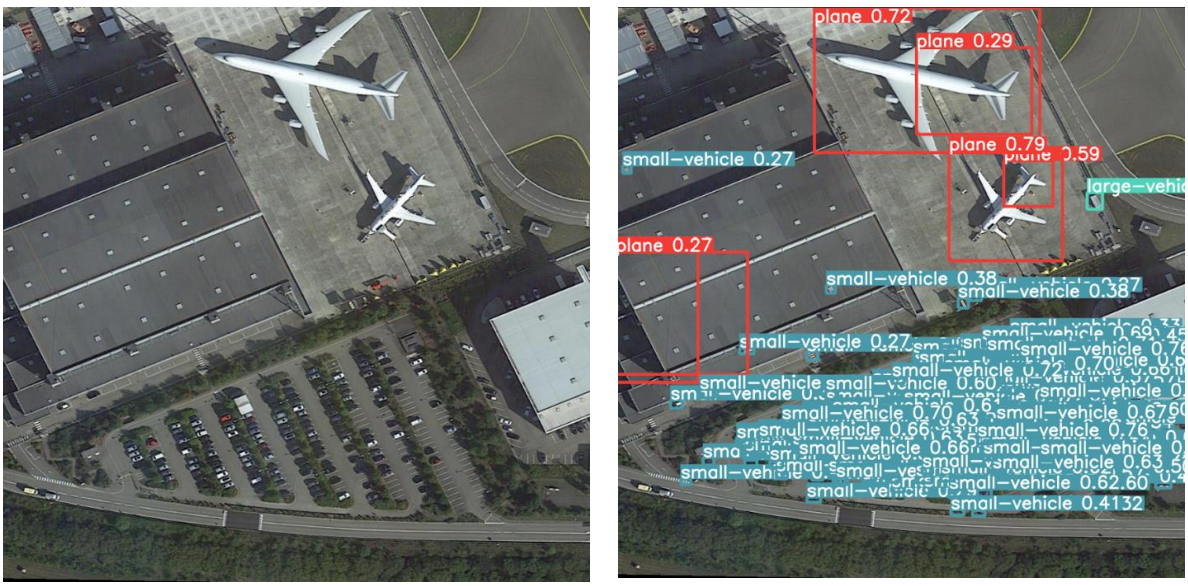


Fig. Loss curve of the YOLO Model

The loss curve shows a gradual decline in the train loss and the validation loss. As the validation loss is not increasing after a certain point, we can infer that the model has not overfit.

Results on testing data



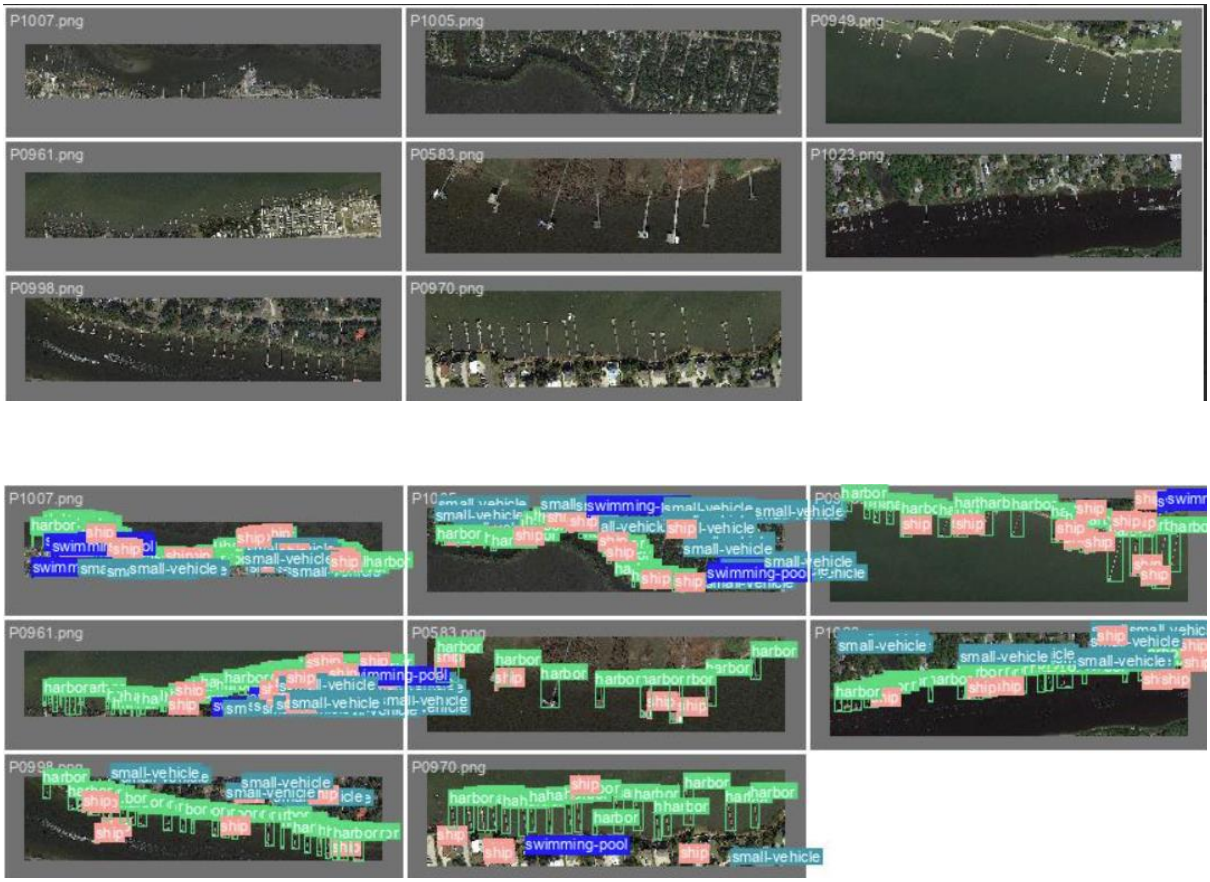


Fig. Input test data and Final image with predictions and bounding boxes

Source Code

```
import cv2
import numpy as np
import matplotlib.pyplot as plt
from dataloader import DataLoader
import os
import glob
from pathlib import Path
from resizing_BB import BB_resize
from splitting_img import splitting
if __name__ == '__main__':
    images_path = 'images2'
    list_images = []
    for filename in os.listdir(images_path):
        file_path = os.path.join(images_path, filename)
```

```
        list_images.append(file_path)

    list_annots = []
    annots_path = 'labels2'
    for filename in os.listdir(annots_path):
        file_path = os.path.join(annots_path, filename)
        list_annots.append(file_path)
    dataloader = DataLoader()
    all_boundingbox_coords, all_labels = dataloader.data(annot_path=annots_path)
    resize = BB_resize()
    split_image = splitting()
    for i, image_path in enumerate(list_images):
        bounding_boxes = np.array(all_boundingbox_coords[i])
        labels_boxes = np.array(all_labels[i])
        image = cv2.imread(f'{image_path}')
        sub_images, sub_boxes, labels = split_image.split(image, bounding_boxes=bounding_boxes,
labels=labels_boxes)
        file_name = image_path[8:13]
        print(file_name)
        for subplot in range(9):
            #img = dataloader.ImageVisualization(sub_images[subplot], sub_boxes[subplot])
            #print(i)
            #print((np.array(sub_boxes[i])))
            res_img, res_box, label_finals = resize.resize(sub_images[subplot], sub_boxes[subplot],
labels=labels)
            res_box = np.array(res_box)
            #print(res_box)
            img = dataloader.ImageVisualization(res_img, res_box)
            #print(f'images_resized\{file_name}_{subplot}')
            cv2.imwrite(f'images_resized3\{file_name}_{subplot}.png', img)
            file_txt = f'{file_name}_{subplot}.txt'
            folder_name = 'bounding_box_resized3'
            with open(os.path.join(folder_name, file_txt), 'w') as file:
                i=0
                for annot in res_box:
                    for item in annot:
                        file.write("%s " % item)
                    file.write("%s" % label_finals[subplot][i])
                    i = i+1
                    file.write("\n")
```

This is the source code for pre-processing stage. When we run this it calls the dataloader.py file, resizing_bb.py file and the splitting.py file. This takes the initial image as the input and divides it into 9 different resized image, with a constant pixel size of 1024x1024.

This data is passed on into the yolo model. The yolo model and all the other files that are needed has also been uploaded into the box folder.

Literature Review

1. Classification

Classification is a fundamental task in remote sensing data analysis, where the goal is to assign a semantic label to each image, such as 'urban', 'forest', 'agricultural land', etc. The process of assigning labels to an image is known as image-level classification. However, in some cases, a single image might contain multiple different land cover types, such as a forest with a river running through it, or a city with both residential and commercial areas. In these cases, image-level classification becomes more complex and involves assigning multiple labels to a single image. This can be accomplished using a combination of feature extraction and machine learning algorithms to accurately identify the different land cover types.

Study -1.1: Remote Sensing Image Classification via Improved Cross-Entropy Loss and Transfer Learning Strategy Based on Deep Convolutional Neural Networks

- This method includes proposing a new loss function, which enhances the focus of the network on hard examples by adding a new term to CE as a penalty term, designing a new multilayer perceptron (MLP) as a classifier, in which the used attention mechanism extracts more discriminative features by weighting each of the channels adaptively; and applying transfer learning strategy by adopting neural architecture search network mobile (NASNet Mobile) as a feature descriptor for the first time in the field of aerial images, which can mitigate the aforementioned costs.

Study -1.2: Patch-Based Discriminative Learning for Remote Sensing Scene Classification

- The proposed method employs multi-level feature learning based on small, medium, and large neighborhood regions to enhance the discriminative power of image representation. To achieve this, image patches are selected through a fixed-size sliding window, and *sampling redundancy*, a novel concept, is developed to minimize the occurrence of redundant features while sustaining the relevant features for the model.
- Apart from multi-level learning, they explicitly impose image pyramids to magnify the visual information of the scene images and optimize their positions and scale parameters locally. Motivated by this, a local descriptor is exploited to extract multi-level and multi-scale features that we represent in terms of a *codeword* histogram by performing k-means clustering.
- Finally, a simple fusion strategy is proposed to balance the contribution of individual features where the fused features are incorporated into a bidirectional long short-term memory (BiLSTM) network. Experimental results on the NWPU-RESISC45, AID, UC-Merced, and WHU-RS datasets demonstrate that the proposed approach yields significantly higher classification performance in comparison with existing state-of-the-art deep-learning-based methods.

Study -1.3: Convolutional Neural Networks Based Remote Sensing Scene Classification under Clear and Cloudy Environments

- In this paper, they propose a new CNN based network (named GLNet) with the Global Encoder and Local Encoder to extract the discriminative global and local features for the RS scene classification, where the constraints for inter-class dispersion and intra-class compactness are embedded in the GLNet training.
- The experimental results on two publicized RS scene classification datasets show that the proposed GLNet could achieve better performance based on many existing CNN backbones under both clear and cloudy environments.

Study -1.4: Deep learning for multi-modal classification of cloud, shadow and land cover scenes in PlanetScope and Sentinel-2 imagery

- In this study, they developed deep learning models able to efficiently and accurately classify cloud, shadow and land cover scenes in different high-resolution (<10 m) satellite imagery.
- Specifically, they trained deep convolutional neural network (CNN) models to perform multi-label classification of multi-modal, high-resolution satellite imagery at the scene level. Multi-label classification at the scene level (a.k.a. image indexing), as opposed to the pixel level, allows for faster performance, higher accuracy (although at the cost of detail) and higher generalizability.
- They investigated the generalization ability (i.e. cross-dataset and geographic independence) of individual and ensemble CNN models trained on multi-modal satellite imagery (i.e. PlanetScope and Sentinel-2). The models trained on PlanetScope imagery collected over the Amazon.

2. Segmentation

Image segmentation is a crucial step in image analysis and computer vision, with the goal of dividing an image into semantically meaningful segments or regions. The process of image segmentation assigns a class label to each pixel in an image, effectively transforming an image from a 2D grid of pixels into a 2D grid of pixels with assigned class labels. One common application of image segmentation is road or building segmentation, where the goal is to identify and separate roads and buildings from other features within an image. To accomplish this task, single class models are often trained to differentiate between roads and background, or buildings and background. These models are designed to recognize specific features, such as color, texture, and shape, that are characteristic of roads or buildings, and use this information to assign class labels to the pixels in an image.

Study -2.1: LoveDA: A Remote Sensing Land-Cover Dataset for Domain Adaptive Semantic Segmentation

- In this paper, they introduce the Land-cOVEr Domain Adaptive semantic segmentation (LoveDA) dataset to advance semantic and transferable learning. The LoveDA dataset contains 5987 HSR images with 166768 annotated objects from three different cities.
- Compared to the existing datasets, the LoveDA dataset encompasses two domains (urban and rural), which brings considerable challenges due to the: 1) multi-scale objects; 2) complex background samples; and 3) inconsistent class distributions.
- The LoveDA dataset is suitable for both land-cover semantic segmentation and unsupervised domain adaptation (UDA) tasks. Accordingly, they benchmarked the LoveDA dataset on eleven semantic segmentation methods and eight UDA methods. Some exploratory studies including multi-scale architectures and strategies, additional background supervision, and pseudo-label analysis were also carried out to address these challenges.

Study -2.2: Large-scale automatic identification of urban vacant land using semantic segmentation of high-resolution remote sensing images

- Criteria inconsistency in cross-domain identification is also a major challenge. To address these problems, the study proposed a large-scale automatic identification framework of urban vacant land based on semantic segmentation of high-resolution remote sensing images and select 36 major cities in China as study areas.
- The framework utilizes deep learning techniques to realize automatic identification and introduces the city stratification method to address the challenge of identification criteria inconsistency.
- The proposed framework provides a practical approach to large-scale vacant land identification in various countries and regions worldwide, which is of great significance for the academic development of urban vacant land and future urban development.

Study -2.3: Land Use and Land Cover Mapping Using Deep Learning Based Segmentation Approaches and VHR Worldview-3 Images

- In this research, they generated a new benchmark dataset from VHR Worldview-3 images for twelve distinct LULC classes of two different geographical locations. They evaluated the performance of different segmentation architectures and encoders to find the best design to create highly accurate LULC maps.
- This design could be used by other researchers for LULC mapping of similar classes from different satellite images or for different geographical regions. Moreover, this dataset can be used as a reference for implementing new segmentation models via supervised, semi- or weakly-supervised deep learning models. In addition, the model results can be used for transfer learning and generalizability of different methodologies.

3. Object Detection

Object detection in remote sensing involves locating and surrounding objects of interest with bounding boxes. Due to the large size of remote sensing images and the fact that objects may only comprise a few pixels, object detection can be challenging in this context. The imbalance between the area of the objects to be detected and the background, combined with the potential for objects to be easily confused with random features in the background, further complicates the task. Object detection generally performs better on larger objects, but becomes increasingly difficult as the objects become smaller and more densely packed. The accuracy of object detection models can also degrade rapidly as image resolution decreases, which is why it is common to use high resolution imagery, such as 30cm RGB, for object detection in remote sensing.

Study -3.1: Axis Learning for Orientated Objects Detection in Aerial Images

- In this article, they propose a new one-stage anchor-free method to detect orientated objects in per-pixel prediction fashion with less computational complexity. Arbitrary orientated objects are detected by predicting the axis of the object, which is the line connecting the head and tail of the object, and the width of the object is vertical to the axis.
- By predicting objects at the pixel level of feature maps directly, the method avoids setting a number of hyperparameters related to anchor and is computationally efficient. Besides, a new aspect-ratio-aware orientation centerness method is proposed to better weigh positive pixel points, in order to guide the network to learn discriminative features from a complex background, which brings improvements for large aspect ratio object detection.
- The method is tested on two common aerial image datasets, achieving better performance compared with most one-stage orientated methods and many two-stage anchor-based methods with a simpler procedure and lower computational complexity.

Study -3.1: Dual-Aligned Oriented Detector

- In this article, they present a two-stage oriented object detection method, termed dual-aligned oriented detector (DODet), toward evading the aforementioned problems of spatial and feature misalignments. In DODet, the first stage is an oriented proposal network (OPN), which generates high-quality oriented proposals via a novel representation scheme of oriented objects.
- The second stage is a localization-guided detection head (LDH) that aims at alleviating the feature misalignment between classification and localization.
- Comprehensive and extensive evaluations on three benchmarks, including DIOR-R, DOTA, and HRSC2016, indicate that the method could obtain consistent and substantial gains compared with the baseline method.

4. Cloud detection & removal

Clouds are a major issue in remote sensing images as they can obscure the underlying ground features. This hinders the accuracy and effectiveness of remote sensing analysis, as the obscured regions cannot be properly interpreted. In order to address this challenge, various techniques have been developed to detect clouds in remote sensing images. Both classical algorithms and deep learning approaches can be employed for cloud detection. Classical algorithms typically use threshold-based techniques and hand-crafted features to identify cloud pixels. However, these techniques can be limited in their accuracy and are sensitive to changes in image appearance and cloud structure. On the other hand, deep learning approaches leverage the power of convolutional neural networks (CNNs) to accurately detect clouds in remote sensing images.

Study -4.1: Using Convolutional Neural Networks for Cloud Detection on VENμS Images over Multiple Land-Cover Types

- The overarching goal of the study is to explore and compare the performance of three Convolutional Neural Network architectures (U-Net, SegNet, and DeepLab) for detecting clouds in the VENμS satellite images.
- To fulfil this goal, three VENμS tiles in Israel were selected. The tiles represent different land-use and cover categories, including vegetated, urban, agricultural, and arid areas, as well as water bodies, with a special focus on bright desert surfaces. Additionally, the study examines the effect of various channel inputs, exploring possibilities of broader usage of these architectures for different data sources. It was found that among the tested architectures, U-Net performs the best in most settings. Its results on a simple RGB-based dataset indicate its potential value for any satellite system screening, at least in the visible spectrum.
- It is concluded that all of the tested architectures outperform the current VENμS cloud-masking algorithm by lowering the false positive detection ratio by tens of percents, and should be considered an alternative by any user dealing with cloud-corrupted scenes.

Study -4.2: Explaining the Effects of Clouds on Remote Sensing Scene Classification

- The study provides a thorough investigation of how classifiers trained on cloud-free data fail once they encounter noisy imagery—a common scenario encountered when deploying pretrained models for remote sensing to real use cases. The paper shows how and why remote sensing scene classification suffers from cloud coverage. Based on a multistage analysis, including explainability approaches applied to the predictions, we work out four different types of effects that clouds have on scene prediction.
- The contribution of this work is to deepen the understanding of the effects of clouds on common remote sensing applications and consequently guide the development of more robust methods.

Study -4.3: Deep Internal Learning for Inpainting of Cloud-Affected Regions in Satellite Imagery

- In this study, cloud removal was implemented within an internal learning regime through an inpainting technique based on the deep image prior.
- The approach was evaluated on both a synthetic dataset with an exact ground truth, as well as real samples. The ability to inpaint the cloud-affected regions for varying weather conditions across a whole year with no prior training was demonstrated, and the performance of the approach was characterised.
- A common approach to cloud removal treats the problem as an inpainting task and imputes optical data in the cloud-affected regions employing either mosaicing historical data or making use of sensing modalities not impacted by cloud obstructions, such as SAR. Recently, deep learning approaches have been explored in these applications; however, the majority of reported solutions rely on external learning practices, i.e., models trained on fixed datasets. Although these models perform well within the context of a particular dataset, a significant risk of spatial and temporal overfitting exists when applied in different locations or at different times.

5. Autoencoders, dimensionality reduction, image embeddings & similarity search

Autoencoders are a type of neural network that aim to simplify the representation of input data by compressing it into a lower dimensional form. This is achieved through a two-step process of encoding and decoding, where the encoding step compresses the data into a lower dimensional representation, and the decoding step restores the data back to its original form. The goal of this process is to reduce the data's dimensionality, making it easier to store and process, while retaining the essential information. Dimensionality reduction, as the name suggests, refers to the process of reducing the number of dimensions in a dataset. This can be achieved through various techniques such as principal component analysis (PCA) or singular value decomposition (SVD).

Study -5.1: Unsupervised Satellite Image Time Series Clustering Using Object-Based Approaches and 3D Convolutional Autoencoder

- In this paper, they propose an algorithm that performs both segmentation and clustering of SITS. It is achieved by using a compressed SITS representation obtained with a multi-view 3D convolutional autoencoder.
- First, a unique segmentation map is computed for the whole SITS. Then, the extracted spatio-temporal objects are clustered using their encoded descriptors. The proposed approach was evaluated on two real-life datasets and outperformed the state-of-the-art methods.

Study -5.2: Estimating generalized measures of local neighbourhood context from multispectral satellite images using a convolutional neural network

- This paper exploits advances in machine learning to implement a new method of capturing measures of urban context from multispectral satellite imagery at a very small area level through the application of a convolutional autoencoder (CAE).
- The utility of outputs from the CAE is enhanced through the application of spatial weighting, and the smoothed outputs are then summarised using cluster analysis to generate a typology comprising seven groups describing salient patterns of differentiated urban context.
- The limits of the technique are discussed with reference to the resolution of the satellite data utilised within the study and the interaction between the geography of the input data and the learned structure. The method is implemented within the context of Great Britain, however, is applicable to any location where similar high resolution multispectral imagery are available.

6. Image retrieval

Image retrieval is the task of retrieving images from a collection that are similar to a query image. Image retrieval plays a vital role in remote sensing by enabling the efficient and effective search for relevant images from large image archives, and by providing a way to quantify changes in the environment over time.

Study -6.1: Remote Sensing Cross-Modal Text-Image Retrieval Based on Global and Local Information

- In this study, they first propose a RSCTIR framework based on global and local information (GaLR), and design a multi-level information dynamic fusion (MIDF) module to efficaciously integrate features of different levels. MIDF leverages local information to correct global information, utilizes global information to supplement local information, and uses the dynamic addition of the two to generate prominent visual representation.
- To alleviate the pressure of the redundant targets on the graph convolution network (GCN) and to improve the model's attention on salient instances during modeling local features, the denoised representation matrix and the enhanced adjacency matrix (DREA) are devised to assist GCN in producing superior local representations. DREA not only filters out redundant features with high similarity, but also obtains more powerful local features by enhancing the features of prominent objects.
- Finally, to make full use of the information in the similarity matrix during inference, we come up with a plug-and-play multivariate rerank (MR) algorithm. The algorithm utilizes the k nearest neighbors of the retrieval results to perform a reverse search, and improves the performance by combining multiple components of bidirectional retrieval. Extensive experiments on public datasets strongly demonstrate the state-of-the-art performance of GaLR methods on the RSCTIR task.

7. Self-supervised, unsupervised & contrastive learning

Self-supervised, unsupervised & contrastive learning are all methods of machine learning that use unlabeled data to train algorithms. Self-supervised learning uses labeled data to create an artificial supervisor, while unsupervised learning uses only the data itself to identify patterns and similarities. Contrastive learning uses pairs of data points to learn representations of data, usually for classification tasks.

Study -7.1: Research on Self-Supervised Building Information Extraction with High-Resolution Remote Sensing Images for Photovoltaic Potential Evaluation

- In this paper, they design a pseudo-label-guided self-supervised learning (PGSSL) semantic segmentation network structure based on high-resolution remote sensing images to extract building information.
- The pseudo-label-guided learning method allows the feature results extracted by the pretext task to be more applicable to the target task and ultimately improves segmentation accuracy.
- The proposed method achieves better results than current contrastive learning methods in most experiments and uses only about 20–50% of the labeled data to achieve comparable performance with random initialization.
- In addition, a more accurate statistical method for building density distribution is designed based on the semantic segmentation results. This method addresses the last step of the extraction results oriented to the PV potential assessment, and this paper is validated in Beijing, China, to demonstrate the effectiveness of the proposed method.

Citations

- Study-1.1: A. Bahri, S. Ghofrani Majelan, S. Mohammadi, M. Noori and K. Mohammadi, "Remote Sensing Image Classification via Improved Cross-Entropy Loss and Transfer Learning Strategy Based on Deep Convolutional Neural Networks," in IEEE Geoscience and Remote Sensing Letters, vol. 17, no. 6, pp. 1087-1091, June 2020, doi: 10.1109/LGRS.2019.2937872.
- Study-1.2: Muhammad, U. et al. (2022) Patch-based discriminative learning for Remote Sensing Scene Classification, MDPI. Available at: https://www.mdpi.com/2072-4292/14/23/5913.
- Study-1.3: H. Sun, Y. Lin, Q. Zou, S. Song, J. Fang and H. Yu, "Convolutional Neural Networks Based Remote Sensing Scene Classification under Clear and Cloudy Environments," 2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), Montreal, BC, Canada, 2021, pp. 713-720, doi: 10.1109/ICCVW54120.2021.00085.
- Study-1.4: Yuri Shendryk, Yannik Rist, Catherine Ticehurst, Peter Thorburn,”Deep learning for multi-modal classification of cloud, shadow and land cover scenes in PlanetScope and Sentinel-2 imagery”,ISPRS Journal of Photogrammetry and Remote

Sensing,Volume 157,2019. Available at: https://www.sciencedirect.com/science/article/pii/S0924271619302023

- Study-2.1: Wang, J. et al. (2022) LoveDA: A Remote Sensing Land-Cover Dataset for Domain Adaptive Semantic Segmentation. Available at: https://arxiv.org/pdf/2110.08733.pdf
- Study-2.2: Lingdong Mao, Zhe Zheng, Xiangfeng Meng, Yucheng Zhou, Pengju Zhao, Zhihan Yang, Ying Long,"Large-scale automatic identification of urban vacant land using semantic segmentation of high-resolution remote sensing images,Landscape and Urban Planning", ISSN 0169-2046, June 2022. Available at: https://www.sciencedirect.com/science/article/pii/S0169204622000330
- Study-2.3: Sertel, E. et al. (2022) Land use and land cover mapping using Deep Learning based segmentation approaches and VHR Worldview-3 Images, MDPI. Available at: https://www.mdpi.com/2072-4292/14/18/4558.
- Study-3.1: Xiao, Z. et al. (2020) Axis learning for orientated objects detection in aerial images, MDPI. Available at: https://www.mdpi.com/2072-4292/12/6/908.
- Study-3.2:G. Cheng et al., "Dual-Aligned Oriented Detector," in IEEE Transactions on Geoscience and Remote Sensing, vol. 60, pp. 1-11, 2022, Art no. 5618111, doi: 10.1109/TGRS.2022.3149780.
- Study-4.1: Pešek, O.; Segal-Rozenhaimer, M.; Karnieli, A. Using Convolutional Neural Networks for Cloud Detection on VENμS Images over Multiple Land-Cover Types. Remote Sens. 2022, 14, 5210. https://doi.org/10.3390/rs14205210
- Study-4.2: J. Gawlikowski, P. Ebel, M. Schmitt and X. X. Zhu, "Explaining the Effects of Clouds on Remote Sensing Scene Classification," in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 15, pp. 9976-9986, 2022, doi: 10.1109/JSTARS.2022.3221788.
- Study-4.3: Czerkawski, M.; Upadhyay, P.; Davison, C.; Werkmeister, A.; Cardona, J.; Atkinson, R.; Michie, C.; Andonovic, I.; Macdonald, M.; Tachtatzis, C. Deep Internal Learning for Inpainting of Cloud-Affected Regions in Satellite Imagery. Remote Sens. 2022, 14, 1342. https://doi.org/10.3390/rs14061342
- Study-5.1: Kalinicheva, Ekaterina & Sublime, Jeremie & Trocan, Maria. (2020). Unsupervised Satellite Image Time Series Clustering Using Object-Based Approaches and 3D Convolutional Autoencoder. Remote Sensing. 12. 10.3390/rs12111816.
- Study-5.2: Alex Singleton, Dani Arribas-Bel, John Murray, Martin Fleischmann,"Estimating generalized measures of local neighbourhood context from multispectral satellite images using a convolutional neural network,Computers, Environment and Urban Systems", ISSN 0198-9715, July 2022, Available at: https://www.sciencedirect.com/science/article/pii/S0198971522000461
- Study-6.1: Z. Yuan et al., "Remote Sensing Cross-Modal Text-Image Retrieval Based on Global and Local Information," in IEEE Transactions on Geoscience and Remote Sensing, vol. 60, pp. 1-16, 2022, Art no. 5620616, doi: 10.1109/TGRS.2022.3163706.
- Study-7.1: Chen, D.-Y.; Peng, L.; Zhang, W.-Y.; Wang, Y.-D.; Yang, L.-N. Research on Self-Supervised Building Information Extraction with High-Resolution Remote Sensing Images for Photovoltaic Potential Evaluation. Remote Sens. 2022, 14, 5350. https://doi.org/10.3390/rs14215350